Author:
Bloomfield, Mark William

Title:
Efficient Optimization of Laminated Composites

Efficient Optimization of Laminated Composites

Mark William Bloomfield

A dissertation submitted to the University of Bristol in accordance with the requirements for award of degree of Doctor of Philosophy in the Faculty of Engineering, Department of Aerospace Engineering, May 2010.

47,000 Words

This thesis is dedicated to my family who has taught me to question everything and never give up. I am who I am because of you.

Acknowledgements

I would like to thank my supervisor, Professor Paul Weaver for giving me the opportunity to undertake my PhD and for allowing me the space and freedom to explore my ideas. Furthermore, his guidance and continuous support over the past three years have been greatly appreciated. I would also like to express my sincere gratitude to Great Western Research and Airbus UK for funding my research without which this thesis would not have taken place.

I would like to express my sincere thanks to Dr. Cezar Diaconu for his initial guidance, advice and encouragement regarding lamination parameters. I also wish to say a special thank you to Dr. Enrique Herencia who provided me with a great deal of support, technical assistance and insight and was always willing to listen to my ideas and answer my questions!

I also wish to thank my colleagues in the Department of Aerospace Engineering for their friendship and support. In particular, Giacomo Canale, Michael May, Alberto Pirrera, Ajit Panesar and Christophe Thill who I have enjoyed working with immensely. I wish you all well in your future careers.

Additionally, I would like to express my heartfelt thanks and gratitude to my parents and family who have always supported my decisions. I can never express how grateful I am for all the sacrifices you made which have allowed me being an 'eternal student'. Finally, I would like to thank my wife Jane for her love, support and encouragement over the past three years. Listening to me talk about lamination parameters, ants and birds cannot be that easy. I love you more than optimization!

Abstract

Currently, there are few computationally efficient optimization tools that provide detailed structural analysis for multi-part laminated composite structures. This shortcoming is due partly to the explosion in number of design variables with an increasing number of parts. To reduce the number of design variables on a local level, lamination parameters, which are trigonometric functions of the laminate stacking sequence, are used. It has been proven that the relationships between lamination parameters form a convex domain. Therefore, if the objective function and constraints are convex, using gradient based methods ensure global optima are obtained. A two-level optimization approach is detailed and used to determine the stacking sequences of composite plates of minimum mass. At the first level, lamination parameters and plate thickness are used to minimize the structural mass subject to buckling, strength (allowable laminate strain) and lamination parameter (feasible region) constraints. A general method is presented to determine the set of constraints on the feasible region of lamination parameters for any finite set of ply orientations. The output of the first level is the minimum thickness. At the second level, ply orientations and a discrete optimizer are used to determine a laminate stacking sequence of that thickness which satisfies the set of design constraints. Formally, this is a constraint satisfaction problem. To solve the second level problem, a number of meta-heuristic techniques are considered including an ant colony and particle swarm approach. Motivated by the analysis, three new solutions to the second level are presented. Additionally, using an expanded set of ply orientations (and a multi-level approach) it is demonstrated that mass savings can be achieved. Furthermore, the presented algorithms lead to efficiency savings in comparison with methods detailed in the literature. Consequently, this thesis presents an efficient and reliable approach to laminated composite optimization.

Author's Declaration

I declare that the work in this dissertation was carried out in accordance with the requirements of the University's Regulations and Code of Practice for Research Degree Programmes and that it has not been submitted for any other academic award. Except where indicated by specific reference in the text, the work is the candidate's own work. Work done in collaboration with, or with the assistance of, others, is indicated as such. Any views expressed in the dissertation are those of the author.

SIGNED:...DATE: 10/05/2010

Table of Contents

List of Figures

List of Tables

List of Abbreviations

ACO	=	Ant Colony Optimization
BB	=	Branch and Bound
CA	=	Cellular Automata
CFS	=	Closed Form Solution
CPSO	=	Combinatorial Particle Swarm Optimization
DBM	=	Direct Branching Method
DIL	=	Distribution in Internal Loads
DOE	=	Design of Experiments
FC	=	Function Count
FEA	=	Finite Element Analysis
GA	=	Genetic Algorithm
GC	=	Geometry Case
HPC	=	High Performance Computing
IP	=	Integer Programming
LBNN	=	Lower Bound Nearest Neighbour
LC	=	Load Case
LP	=	Linear Programming
MILP	=	Mixed Integer Linear Programming
MINLP	=	Mixed Integer Non-Linear Programming
MLM	=	Maximum Lagrange Multiplier
MP	=	Mathematic Programming
MPSO	=	Modified Particle Swarm Optimization
NI	=	Number of Iterations
PS	=	Ply Set
QP	=	Quadratic Programming
PSO	=	Particle Swarm Optimization
RBO	=	Reliability Based Optimization
RF	=	Reserve Factor
RSM	=	Response Surface Methodology
SDGD	=	Stochastic Discrete Gradient Descent
SQP	=	Sequential Quadratic Programming
UBNN	=	Upper Bound Nearest Neighbour

Nomenclature

θ_i	=	The ith ply orientation
$\xi_i^{A,B,D}$	=	Lamination parameters (i = 1...4)
ϕ	=	Set of ply orientations
$\Gamma, \hat{\Gamma}$	=	Vector of lamination parameters
p_{ij}	=	Ant probability table
α_{ij}	=	Ant routing table
$\overline{h}_i^j$	=	Coefficients of hyperplane constraints
h_i^j	=	Coefficients of normalised hyperplane constraints
N_i^{cr}	=	Critical load per unit width
ρ	=	Density of material
τ_{ij}^k	=	Pheromone deposited by ant k on arc ij
$\bar{\rho}$	=	Pheromone evaporation coefficient
ν_{12}	=	Poisson's ratio
RF_i	=	Reserve factor for strength ($i = s$) and buckling ($i = b$)
ψ_i	=	Scalar quantities
ε^0	=	Vector of mid-plane strains
κ	=	Vector of plate curvatures
E_{11}, E_{22}	=	Longitudinal and transverse Young's moduli
$\alpha_i, \beta_i, \chi_i$	=	Real valued scalars
ω	=	Swarm inertia coefficient
A	=	Cross sectional area of the plate
a	=	Plate length
A_{ij}	=	Components of the in-plane stiffness matrix
B_{ij}	=	Components of the coupling stiffness matrix
b	=	Plate width
C	=	Compression
$c_2(i)$, $c_4(i)$	=	$\cos 2\theta_i$, $\cos 4\theta_i$
c_i	=	Weighting factors
D_{ij}	=	Components of the out-of-plane stiffness matrix
E_{11}, E_{22}	=	Longitudinal and transverse Young's moduli
F	=	Fitness function
G_{12}	=	Shear modulus
G	=	Constraint
G^{eq}	=	Equality constraints

G^{in}	=	Inequality constraints
h	=	Plate thickness
H_j	=	Hyperplane constraints
H^L, H^U	=	Lower and upper bound values of the hyperplane constraints
k_i	=	Constraint scaling factor
m	=	Size of ϕ
M	=	Vector of out-of-plane moments
MS	=	Mid-plane symmetric
n	=	Dimension of the in-plane, coupling or out-of-plane feasible region
N	=	Vector of in-plane loads
n_c	=	Number of design constraints
np	=	Total number of plies in the lay-up
p	=	Number of plies in a stacking sequence
Q_{ij}	=	Components of the reduced stiffness matrix
r	=	Half the number of the plies in the lay-up
$r_1(t), r_2(t)$	=	Random weighting factors
s	=	Symmetric
$s_2(i)$, $s_4(i)$	=	$\sin 2\theta_i$, $\sin 4\theta_i$
t	=	thickness
T	=	Tension
U_i	=	Material invariants
v	=	Vector of lamination parameters
$v_{ij}(t)$	=	Particle swarm velocity vector at time step t
V	=	Set of vectors of lamination parameters
$x_{ij}(t)$	=	Particle swarm position vector at time step t
z_i	=	Normalised through thickness co-ordinate
ξ	=	Random vector of lamination parameters

Chapter 1
Introduction

1.1 Motivation

Driven by environmental and economic targets, there is a greater need for low structural weight in civil and military aircraft. As such, the aviation industry is rapidly employing composite materials for primary structures such as wings and fuselages. This is seen with their deployment in the Airbus A350 XWB as well as the Boeing Dreamliner 787. The excellent performance of composite materials has been well publicised in recent years. Studies have shown they possess excellent stiffness and strength properties. Despite their insertion in high profile products, significant efficiency and functionality gains can be achieved by undertaking stacking sequence (lay-up) optimization.

In lay-up optimization, the design variables are generally ply thickness and ply orientation. In many practical applications, ply thickness is fixed and ply orientations take a range of discrete values. Traditionally, the set of ply orientations is fixed to [0,90,±45] degrees. This restriction is generally due to manufacturing and certification limitations. However, recent research has highlighted the potential benefits of exploiting an expanded set of ply orientations. In particular, 60 degree plies have been proven to be highly beneficial for shear buckling in long rectangular laminated composite plates (Weaver 2006). Using an expanded set of ply orientations generally increases the complexity of the optimization as well as the manufacturing process. Despite these potential issues, this thesis focuses on the potential gains of a multi-level optimization of laminated composites using an expanded set of ply orientations using lamination parameters.

Currently, there are practically no computationally efficient optimization tools that provide detailed structural analysis for multi-part laminated composite structures. Previous attempts have been made using gradient and/or meta-heuristic optimization approaches. Despite initial success, their limitations soon became apparent. This is partly due to the increasing number of design variables with increasing thickness. In contrast, recent focus has been on the optimization of laminated composite structures using lamination parameters (Herencia et al. 2007a, Kameyama and Fukunaga 2007, Setoohdeh et al. 2006,). Lamination parameters, which are trigonometric functions of the laminate stacking sequence, are particularly useful intermediate design variables in stacking sequence optimization. Furthermore, lamination parameters offer a convenient and manageable route for laminated composite design optimization. This is because the maximum number of parameters is twelve. This is particularly important when the composite designer uses an expanded set of ply orientations. In contrast, using ply orientation and ply thickness as design variables yields potentially infinite combinations. Note, in this thesis, the total laminate thickness is a variable in the optimization instead of the individual ply thicknesses.

The feasible region of lamination parameters, i.e. the parameterized space containing all possible stacking sequences, was proven to be convex (Grenestedt and Gudmundson, 1993) As such, where the objective functions and constraints are convex functions of the lamination parameters, global optima are obtained. In contrast, using ply orientations as design variables leads to a non-convex response surface which may yield local optima when using gradient based methods. It should be noted that whilst the problems presented in this thesis are convex in the lamination parameter space, the introduction of thicknessess in multi-part laminated composite design generally results in a non-convex problem. The feasible region of lamination parameters is currently known only for small limiting sets of ply orientations, or for unique combinations of lamination parameters. Motivated by this, this thesis seeks to determine a general analytical approach to compute the feasible region of lamination parameters for any finite discrete set of ply orientations. Once the feasible region is determined the structural optimization can be undertaken using lamination parameters as an effective parameterization. Ultimately, the optimization will reduce the structural weight of the composite plate/structure by increasing the design space envelope and ensuring accurate representation of the laminate stiffness characteristics via lamination parameters. It should be noted that the feasible region combining the number of each ply orientation and the out-of-plane lamination parameters is non-convex.

Driven by the need to have an effective approach for single and multi-part laminated composite structures, a two-level optimization approach is adopted. At the first level, lamination parameters and plate thickness are used to minimize the mass of the plate subject to strength (laminate allowable strain) buckling and feasible region (lamination parameter) constraints. Once the minimum mass (and hence thickness) is obtained, a discrete optimizer is used to identify a stacking sequence which satisfies the set of constraints. Traditionally, the second level has been solved using a genetic algorithm (GA). However, the thesis addresses alternative approaches to solving the second level problem motivated by the necessity to increase efficiency and reliability. In particular, a particle swarm and ant colony optimizations are considered. Furthermore, the standard models will be enhanced to improve the efficiency, robustness and reliability.

As this thesis concerns the derivation of a framework for the efficient optimization of lamination composites, the approach is discussed with reference to multi-part laminated composite structures via a decomposition method. A conceptual approach is presented where the optimization is undertaken in parallel to increase the efficiency of the optimization process. After an introductory literature review, followed by research objectives and summary of chapters, the main body of this thesis is presented.

1.2 Literature Review

In this section, a formal literature review, which motivates the research presented in this thesis, is given.

1.2.1 Introduction

The following literature review is separated into five key areas:

1. Optimization of Laminated Composites Using Lamination Parameters
2. Strength and Buckling Design Constraints

3. Topological Optimization Using Tow-Steering of Fibres
4. Parallel Optimization of Composite Structures
5. Optimization of Laminated Composites Using Polar Parameters

The motivation behind the review is to provide context to this thesis. Additionally, the literature review is supplemented during the course of this thesis where necessary to provide context and insight.

Whilst formal optimization theory is not reproduced in significant detail, where necessary, references are provided. Nonetheless, it is important to note that optimization has evolved significantly over the past 300 years. From Gauss (Nocedal and Boyd - 1999) who developed a steepest descent gradient approach to Dantzig (1953) who pioneered an algorithmic approach to linear programming – the diversity of breakthroughs has been immense. Current research is focused on large-scale algorithms (such as parallel optimization), global optimization, sequential quadratic programming (SQP) and mixed-integer non-linear programming (MINLP), that is solutions to non-linear problems where the variables can take discrete or continuous values. It is observed that the optimization problems presented in this thesis fall into this particular complex class of problems. Over the past 25 years, optimization theory has been successfully applied to laminated composite structures and is now the cornerstone in the design and improvement of composite structures. In section 1.2.2, literature concerning the optimization of laminated composites using lamination parameters is presented.

1.2.2 Optimization Using Lamination Parameters

Tsai et al. (1962) & Tsai and Hahn (1980) characterized the stiffness properties of composite laminates in terms of material invariants and at most 12 lamination parameters. Lamination parameters were first used in laminate design optimization by Miki (1982). The authors used lamination parameters to determine stacking sequences (of fixed thickness) which maximized buckling load. These designs were obtained utilising the geometric relationships between the feasible region of lamination parameters and the objective function (which was expressed explicitly in terms of lamination parameters).

Fukunaga and Vanderplaats (1991) used lamination parameters and mathematical programming (MP) techniques to perform stiffness optimization of orthotropic laminates. They also attempted to derive the full feasible region of lamination parameters for symmetric orthotropic laminates. The feasible region determined by the authors was in-fact an inner bound to the true feasible region as shown by Grenestedt and Gudmundson (1993). Haftka and Walsh (1992) and Nagendra et al. (1992) used integer programming (IP) techniques and lamination parameters to optimize lay-ups subject to buckling and buckling & strength constraints respectively. Note, most examples focused on laminated with fixed thickness. It is further noted that IP techniques are often computationally expensive (in laminated composite design optimization). Therefore, alternative methods may be sought or alternatively parallel computation may be used to increase the efficiency of the optimization approach. However, it is generally accepted that MP techniques are a compromise between efficiency and scope of task (to a range of small and large scale problems). Note MP/gradient based techniques are not suitable for problems where design variables take discrete values. This is because gradients cannot be computed at discrete points. Fukunaga et al. (1995) used MP techniques and lamination parameters to maximise buckling loads of symmetric laminates under combined loading. This investigation showed that under shear and combined loading, flexural anisotropy could increase or indeed decrease the critical buckling load. Additionally, Grenestedt (1991) performed lay-up optimization of shear panels with and without flexural anisotropy under buckling loads. The authors restricted their study to the out-of-plane lamination parameters only and showed that 60 degree plies were optimal for shear buckling of long thin plates.

Diaconu and Sekine (2004b) performed lay-up optimization of laminated composite shells to maximize the buckling load. Lamination parameters were used as design variables and the feasible region was derived and used as a design constraint in the optimization. Diaconu and Sekine (2004b) derived explicit relationships relating the in-plane, coupling and out-of-plane lamination parameters for angles restricted to [0,90,±45] degree plies. Whilst it is noted that Diaconu and Sekine (2004b) did not provide an explicit method to determine the constraints on the feasible region for the in-plane, coupling and out-of-plane lamination parameters, respectively, they were the first to use the feasible region computed for a finite set of ply orientations. Interestingly, constraints derived on the feasible region of lamination parameters by Diaconu and Sekine (2004b) are consistent with those derived by Liu et al. (2003) (defined for two in-plane and out-of-plane lamination parameters with angles restricted to [0,90,±45] degrees). Note, the expressions were not derived in the manner of Diaconu and Sekine (2004b) but through consideration of the volume fractions of the plies. Tsai and Hahn (1980) showed that the lamination parameters may be expressed in terms of volume fractions. In particular, volume fractions are directly proportional to the in-plane lamination parameters. Thus, it follows that percentages (percentage of each ply orientation in the laminate) are directly related to the in-plane lamination parameters (where the out-of-plane lamination parameters give real stacking sequences, since the in-plane lamination parameters are independent of the stacking sequence). This allows the current industrial terminology of percentages to be converted into more manageable, and at most 12, lamination parameters. Furthermore, Liu et al. (2003) showed that when grouping certain plies, or placing restrictions on the volume fractions of particular ply angles, the feasible region decreases. Consequently, it is observed that such restrictions act as constraints and may affect the optimal solution. Further literature concerning the feasible region of lamination parameters is presented in Chapter 2.

De Visser (1999) adopted a multi-level approach for the optimization of panels in wing structures. At the first level, the in-plane stiffness of the stiffened panels is maximized subject to strength and aeroelastic constraints. At the second level, De Visser (1999) optimizes the panels with respect to buckling load including the in-plane stiffness requirements determined at the first level. De Visser (1999) defines the flexural lamination parameters, not in the same manner as Diaconu and Sekine (2004), but rather in the Liu et al. (2003) fashion. De Visser defined five flexural lamination parameters in terms of the relative volume fractions that each ply contributes to a stacking sequence.

The first lamination parameter is equal to one (the sum of the volume fractions). The fifth lamination parameter is equal to zero, since for any lay-up with angles restricted to [0,90,±45], is identically zero. Furthermore, De Visser derived bounds (upper and lower) on the flexural lamination parameters in terms of the thicknesses of the layers. When the thicknesses take particular values, these upper and bounds become physical constraints on the lay-up. Note, the feasible region determined by Diaconu and Sekine (2004) places no such restrictions on the volume fractions of the ply angles in the lay-up and allows for a more general formulation with the assumption that all constraints can be expressed in terms of lamination parameters. In contrast, the derivation of feasible region constraints such as those presented by Liu et al. (2003) would be difficult to calculate for an expanded set of ply orientations.

In contrast to MP based approaches, meta-heurisitc search techniques like GAs algorithms have also been applied to numerous stacking sequence optimization problems. As previously stated, GAs are probabilistic search techniques based on Darwinian evolution. Unlike MP, GAs do not require gradient information and thus can be applied to problems where there may exist many local optima, i.e. problems with a non-convex objective function. Nagendra et al. (1992) used GAs for the optimization of blade stiffened composite panels. A combined multi-level approach using MP techniques, GAs and lamination parameters was initially proposed by Yamazaki (1996). The optimization was divided into two levels. At the first, level, optimum lamination parameters were determined using MP techniques and lamination parameters were used as design variables. At the second level, a GA was used to target the optimum lamination parameters and match real lay-ups to these parameters. In this approach, buckling loads were maximized. Todoroki and Haftka (1997) furthered the work of Yamazaki (1996). At the first level, lamination parameters were used to identify the neighbourhood of the optimum design. Next, a response surface approximation was created in that neighbourhood. The GA was then used to find real lay-ups that matched the lamination parameters in the response area (in a least squares sense). Recently, Liu et al (2006) used VICONOPT (see Liu et al. 2006) to perform the optimization of a composite stiffened panel subject to strength, buckling and practical design constraints. A two level approach (not incorporating lamination parmeters) was adopted. VICONOPT was used at the first level as a structural analysis tool to minimise weight of the structure subject to constraints. At the second level, GAs was used to determine lay-ups (where thicknesses were rounded up from the first level).

A similar two level approach has been adopted by Herencia et al. (2007a-c, 2008a) to optimize long anisotropic laminated fibre composite panels with T-shape stiffeners. Firstly, gradient based methods are used to minimize the structural mass subject to a set of design constraints (including feasible region constraints). It is observed that the authors restricted their studies to $0, 90, \pm 45$ degree plies. At the second level a GA is used to determine a feasible laminate stacking sequence by targeting the optimum lamination parameters. Recently, Herencia et al. (2008b) modified the second level objective by transforming the problem into a constraint satisfaction problem (CSP). That is, find a lay-up which satisfies a first order approximation of the design constraints. The shift in the second level objective was motivated by the observation that it was difficult to match the optimum lamination parameters and as such additional plies were added at the second level in order to match the lamination parameters leading to conservative results. It is worth noting that the second level of the optimization is highly efficient because no structural analysis is done (i.e. there is no expensive finite element analysis). As such, this second level is rapid and computationally efficient. Although a GA has been used as a successful discrete optimizer at the second level of a bi-level optimization approach, it is worth noting that other methods may yield significant efficiency and functionality improvements. Next, a brief review of the structural constraints used in optimization process is given.

1.2.3 Strength and Buckling Constraints

In any optimization problem, the feasible region captures all design configurations. However, it is important to note there are often other constraints placed upon the design which creates a reduced feasible domain. With respect to laminated composite design, such constraints may include strength (laminate and ply) and buckling. Often strength constraints are introduced to limit the magnitude of the strains in tension, compression and shear. This can be achieved at laminate and/or ply level. Nagendra et al. (1993) integrated strength constraints into the optimization. The authors achieved this by relating ply strength via a strength ratio to the maximum allowable strain a given layer. Additionally, Herencia et al. (2007b, 2008a) successfully applied laminate and ply level strength constraints using allowable strains in lamination parameter space. In this thesis, strength constraints are modeled as allowable strains at a laminate level. This formulation is detailed in Chapter 3.

Kogiso et al. (2003) used reliability based optimization (RBO) with lamination parameters and the Tsai-Wu criteria in strain space. In particular, reliability maximization for a fixed thickness plate as well as a minimization of thickness subject to a reliability constraint. The authors clearly showed that each possible ply orientation created a failure envelope and that for a given set of ply orientations the resulting failure envelope was the intersection of all failure envelopes creating a convex feasible domain in lamination parameter space. More recently, Ijsselmuiden et al. (2007) proposed the use of the Tsai-Wu failure criterion in lamination parameter space. By doing so, the authors created a conservative failure free envelope (in terms of a quadratic and quartic polynomials) completely independent of ply orientation. Note, an envelope can be defined for a single ply orientation. Observing that the failure envelope was convex, the entire envelope was obtained by calculating the convex hull of each failure envelope corresponding to a unique ply orientation. It is important to note that this failure envelope is material dependent. However, this causes few problems as most current design optimization problems (in both academia and industry) concern monolithic (composed of a single material) lay-ups. In sum, whilst this approach is promising, it should be noted that this failure envelope may account for only one particular failure mode. Additionally, if an alternative failure criterion is used, this approach may not be appropriate. As such, knowledge of the full design space in terms of lamination parameters is observed as the most effective route as all other constraints create a feasible sub-space.

Next, buckling constraints in lamination parameter space are considered. Buckling constraints are often assessed at two levels: local and global. With respect to local buckling, this can be considered as the buckling (as shown, for example, by Herencia et al. (2008a)) of a single plate or plate a plate element in a multi-part structure. Global buckling, in contrast, can be seen as the failure of the entire structure. Herencia et al. (2007a-c, 2008a) analysed the structural buckling using the closed form solutions derived by Weaver (2006). Whilst the buckling constraints derived by Weaver are discussed in detail in Chapter 3, they are, for reasons of completeness, outlined here. Weaver (2006) provided a set of closed form solutions for the buckling of long anisotropic plates subject to compression and shear loading. The derived closed form solutions were in terms of non-dimensionalised parameters and built upon the work of Nemeth (1986). These non-dimensionalised parameters were expressed in terms of the components of the out-of-plane stiffness matrix. Hence the non-dimensionalised parameters are functions of the out-of-plane lamination parameters. It should be noted that these closed form solutions provide useful insight into the flexural behaviour of the composite laminate. Furthermore, these closed form solutions are useful from a design perspective and as demonstrated by Herencia et al (2007a-c, 2008a,) useful for lay-up optimization as it increases the efficiency of the procedure as there is less dependency on computationally expensive finite element routines.

Lastly, it is important to note the importance of Finite Element Analysis (FEA) in structural analysis. It is generally accepted, that FEA is one of the most computationally expensive tasks in the optimization process. In itself, this cost is a constraint on any optimization of any complex structure. The use of parallel computation can be used to overcome this limitation and is discussed later. Next, the optimization of tow-steered composite fibres is discussed.

1.2.4 Topological Optimization (tow-steering) of Composite Fibres

In general, topological optimization refers to the optimization of structural topology/shape. A great deal of work has been undertaken with respect to shape and size optimization of wing structures such as ribs, spars etc (Krog et al. 1999) However, in the context of this literature review, topological optimization refers to the optimization of variable stiffness laminates. Traditionally, fibre orientation is constant for each layer in the lay-up. The concept of variable-stiffness laminates allows the stiffness properties to vary spatially over the laminate. Through the optimal tailoring of fibre orientations spatially, the stiffness properties of laminte properties may be vastly improved. This is achieved by using curvilinear or tow-steered fibres.

The study of the effects of curvilinear fibres in composite laminates has evolved over the past 40 years. In 1969, Sendeckyj (1971) showed that curvilinear fibres have a higher longitudinal shear modulus then circular fibres. Hyer and Charette (1991) detailed the gains in structural efficiency by using curvilinear composite laminates. The authors considered a plate with a circular hole (i.e. rib). Generally, a plate with a cut out, such as a wing rib, experiences a high stress concentration around the cut out. By using curvilinear fibres, the authors showed that fibres could 'steer' the stress away from the cut out and thus the stress could be distributed more accordingly across the laminate; a highly attractive possibility. Hyer and Lee (1991) highlighted the increase in buckling resistance of composite plates with circular central holes by using curvilinear fibres. Later, Gürdal and Olmedo (1993) formally introduced the concept of variable stiffness, which is where the stiffness response of a composite laminate may vary point to point. More recently, Setoohdeh at al. (2006) used the feasible region of lamination parameters to optimize the in-plane stiffness of a composite laminate with curvilinear fibres. Setoohdeh et al. (2003) noted that the anisotropic advantageous of fibre reinforced composites may not be fully exploited unless the fibres are properly placed in their optimal spatial orientations. Setoohdeh et al. (2003) investigated the use of cellular automata (CA) for the design and optimization of curvilinear composite fibres for in-plane response. To summise, CA uses local rules to update design variables in an iterative scheme until convergence is achieved. More recently, Setoohdeh at al. (2006) presented a method to determine the curvilinear fibre paths based upon the distribution of optimum lamination parameters. Initially, the optimum distribution of lamination parameters was determined using FEA. Secondly, curve fitting techniques were used to obtain continuous fibre paths matching the distribution of lamination parameters. The main feature of this approach is that at this second level, no expensive finite element analysis is required. Moreover, curve fitting is performed using simple polynomial and trigonometric function evaluations. Setoohdeh et al. (2006) solved the curve fitting problem using a constrained nonlinear least square solver where maximum curvature (due to manufacturing restrictions) is constrained. Setoohdeh et al. (2006) demonstrated the efficiency of this approach by minimising the complementary energy of a plate under normal loading.

In summary, the recent work on curvilinear composite laminates has shown that by using curvilinear composite laminates, the full exploitation of composites anisotropy is achievable. This could be significantly important for elastic tailoring. However, it is noted that this approach is not pursued in this theiss due to current industrial requirements in addition to the manufacturing complexity associated with curvilinear fibres. In the next section, the computational cost associated with FEA is discussed. This culminates in a review of recent work regarding parallel optimization of multi-part composite laminates.

1.2.5 Parallel Optimization of Laminated Composites

In the history of Finite Element Analysis (FEA), demand has always exceeded capabilities. Furthermore, is generally accepted that the computational cost associated with FEA is one of the greatest hindrances to composite structural optimization. This computation time significantly increases as the complexity or size of the structure grows. Whilst closed formed solutions offer invaluable insight and help to reduce the amount of FEA processing, the current trend in research and application is towards the use high performance computation (HPC) in FEA.

HPC can be applied to the design and optimization of composite laminates. Concerning FEA, parallel computation can be used to run an analysis (and/or sensitivity analysis) in a quicker amount of time without loss of accuracy. This allows for the analysis of more complex structures such as wing boxes or entire wing structures. At the FEA level, structural analysis programmes such as MSC Nastran incorporate the superelements function (Patel, 1992). Using superelements, the initial size of the problem is reduced into smaller segments, which can be computed independently. The independence allows the decoupling of a large structural problem into small sub problems. These sub problems require significantly less computation time and often avoid potential 'crashes' due to the exceeding of memory allocation. HPC allows each of the sub problems to be run on a separate processor. In summary, this approach reduces the running time of the individual analyses without a compromise in accuracy.

As noted, parallel computation can be used to reduce the processing time for FEA of complex or large structures. Furthermore, parallel processing can also be applied to direct and heuristic search techniques. Henderson et al. (1994) studied the use of a parallel genetic algorithm for stacking sequence optimization of a composite laminate under buckling, strain and ply contiguity constraints. The authors presented two separate parallelisation schemes. Briefly, parallel genetic algorithms were used to search the design space. The searches took place independently. Once the each GA had converged, each optima from each GA was assessed accordingly. The best local optima was determined from the minimum value of all local optima determined by running the parallel GA. Whilst this approach increases the likelihood of determining the global optima, no guarantees can be made. The method used by Henderson et al. (1994) was slightly different to that of Punch et al. (1994) Punch et al. (1994) proposed the design and optimization of composite structures using coarse-grain parallel genetic algorithms. A coarse-grained system is where is there is infrequent communication, allowing large amounts of computation to take place before data is shared. The authors implemented a new coarse-grain parallel architecture for genetic algorithms, names island injection genetic algorithms. This approach fine-tunes each generation of the population in the GA. The authors showed that by using a parallel GA with an injection algorithm, a speed up in the optimization was achieved

More recently, parallel computation has been successfully implemented using cellular automata (CA) to optimize a variable stiffness plate for in-plane response (Setoohdeh et al. 2006). However, there has been very little work regarding the parallel optimization of composite laminates using lamination parameters for wing substructures. Motivated by this, a conceptual framework for the optimization of multi-part composite laminates using lamination parameters will be presented (Chapter 6).

One of the drawbacks concerning current methods (for the optimization of laminated composite structures) is that for large or complex structures the computational cost associated with FEA is high. Parallel processing is proposed which could be used at the first and second level of the aforementioned optimization strategy. At the first level, for a large or complex structure, FEA can be performed in parallel using distributed computing (such as Kere and Lento 2005), utilizing, for example, the MATLAB distributed computing toolbox. At the second level, a parallel discrete optimizer can be used to speed up the process of determining stacking sequences which satisfy sets of design constraints. This study is very important for future work and for the optimization of large structures. Details concerning parallel optimization will be presented in Chapter 6.

For a monolithic (single material) lay-up, it has been shown in this literature review that lamination parameters offer attractive approach to the optimization of a composite laminate. Moreover, lamination parameters have been used successfully in the optimization of variable stiffness panels for in-plane response. However, if the lay-up is not monolithic but a hybrid, lamination parameters are insufficient to fully characterise the stiffness characteristics of the composite. In the next section, an alternative method is presented which has been successfully used for several optimization problems. Due to its underlying theory, 'The Polar Method' can be used to model the stiffness behaviour hybrid and well as monolithic composite laminate.

1.2.6 Optimization of Laminated Composites Using the Polar Method

Thus far, only lamination parameters have been considered. In this section, a more general formulation in terms of polar parameters is presented. Using polar parameters (Vanucci and Verchery 2001) as design variables, Vincenti et al. (2001) have shown that all elastic tailoring design problems can be formulated as a non-convex optimization problem. Additionally, the objective function of each particular problem considered by the authors can be written explicitly as a positive semi-definite quadratic function of 18 variables (six polar parameters each for the in-plane, coupling and out-of-plane stiffness matrices respectively). This formulation allows a complete general approach to the problem, as no simplifying hypotheses are made. Note, as each optimization problem is non-convex in terms of polar parameters (Vanucci 2005), a gradient based approach cannot guarantee that global minima will be determined. This is in direct contrast to lamination parameters, which have been shown to be an effective approach for laminate composite design,

Vannucci (2005) successfully used the polar method for the analysis and optimization of laminated composite plates including the maximisation of buckling loads. Furthermore, the polar method has been used to define and search for laminates without extension-bending coupling. It was shown by Vannucci (2001) that a particular sub-class of ply orientations, notably anti-symmetric angle ply, in a certain lay-up induced a zero coupling matrix. Traditionally, a symmetric stacking sequence is used to eliminate all extension-out-of-plane coupling. This subclass of lay-ups has the same property, that is, a zero extension-out-of-plane coupling. Interestingly, this could be important for optimization where weight is the primary driver. Furthermore, the polar method has been used successfully to assess the influence of the geometric and mechanical parameters on the loss of elastic properties, such as uncoupling and quasi-homogeneity of laminates composed of identical plies, due to manufacturing errors in ply orientation. In particular, Vincenti et al. (2001) have shown that the influence of the material properties is given by the ratio of two polar parameters, namely, $\frac{R_0}{R_1}$. For further details, see Vincenti et al. (2001).

In summary, the polar method presented by Vanucci and Verchery (2001) has yielded a mathematical and practical insight into laminate composite design. Additionally, the authors have shown that at most 18 polar parameters fully characterise the behaviour of a monolithic as well as hybrid composite. These 18 design variables have been shown to have a non-convex design space (Vanucci 2005). In direct comparison, lamination parameters have at most 12 design variables and have a convex design spaces for monolithic designs. For a monolithic lay-up, whilst noting the insight the polar method has given, lamination parameters offer a more convenient and manageable approach to optimization. As such, this thesis will concern lamination parameters only.

1.3 Research Objectives

Motivated by the literature review and building upon the aforementioned developments in composite optimization, the research objectives are defined as,

1) Derive a general formulation to determine the feasible region of lamination parameters for any finite discrete set of ply orientations
2) Identify suitable methods for continuous optimization using lamination parameters
3) Analyse and identify efficient, reliable and robust discrete optimizers for determining lay-ups in a two-level environment
4) Improve current optimization methods to develop novel approaches for stacking sequence identification
5) Develop a framework for efficient and scalable optimization of laminated composite structures
6) Demonstrate the advantages of points 1-4 through a number of numerical examples

In the next section an outline of the Chapters presented in this thesis is given. Each Chapter captures at least one of the above thesis objectives.

1.4 Outline of Chapters

Chapter 2 details a two-level approach to explicitly derive the feasible regions of lamination parameters. The method details how to calculate the feasible region for in-plane, coupling and out-of-plane lamination parameters as well as the explicit expressions linking all 12 lamination parameters. The detailed approach is valid for finite sets of ply orientations.

Chapter 3 formalises the optimization problem and the two-level approach employed to solve it. Specifically, a gradient based method is used at the first level to identify optimum lamination parameters and thicknesses. At the second level, a discrete optimizer is used to determine a stacking sequence which satisfies the set of constraints.

In **Chapter 4**, a detailed analysis comparing genetic algorithms, particle swarm and ant colony optimization is given. The formal and numerical analysis highlights strong similarities between the approaches. However, the differences between the methods become apparent and conclusions are made.

Chapter 5 builds upon Chapter 4 and several new approaches to stacking sequence optimization are given. Specifically, a modified particle swarm is presented as well as a new combined ant colony direct branching method. Additionally a new stochastic discrete gradient descent approach is detailed. The efficiency and functionality merits of these approaches are highlighted.

Chapter 6 concerns the parallel optimization of multi-part laminated composite structures. A two-level method is adopted, building upon the work presented in this thesis. At the first level, a gradient based method is used to minimise the mass of a idealised wingbox with the sensitivity analysis conducted in parallel. At the second level, a discrete optimizer is used to determine lay-ups, in parallel, which satisfy the set of constraints. The focus of this particular Chapter is the conceptual approach rather than numerical examples.

In **Chapter 7** numerical examples are given concerning mass minimisation of a composite plate subject to strength, buckling and lamination parameter (feasible region) constraints. Additionally, a number of methods are used to determine lay-ups which satisfy the set of constraints where the laminate thickness is determined from the first level. The performance of these methods is analysed.

Finally, in **Chapter 8** conclusions are drawn, reference is given to the work presented in this thesis and its contributions to the field of laminated composite optimization. Additionally, suggestions for future work are also provided.

In the next chapter, a method to determine the feasible region of lamination parameters for finite sets of discrete ply orientations. The research presented in the next Chapter provides the backbone for this thesis.

Chapter 2

On Feasible Regions of Lamination Parameters

2.1 Introduction

The objective of this Chapter is to expand current theory on the feasible region of lamination parameters. After a description of lamination parameters including their characteristics and usage, a two level approach is proposed and used to derive the explicit expressions describing the feasible region of lamination parameters where ply orientations are a predefined finite set. The first level determines, separately, the feasible region of the in-plane, coupling and out-of-plane lamination parameters using convex hulls. It will be shown that the respective feasible regions are enclosed by sets of hyperplane constraints. The second level, using data from the first level, establishes relationships between the hyperplane constraints determined at the first level. As such, feasible regions may be viewed as essentially three separate four dimensional (four lamination parameters) spaces with interconnections between them provided by the second level. This general approach yields all relationships, or constraints, between lamination parameters and thus determines the entire feasible regions of lamination parameters for predefined, finite sets of ply orientations. The derived expressions can then be used to undertake the optimization of laminated composite structures using lamination parameters which is discussed in Chapter 3 onwards.

2.2 Background

Using the classical laminate plate theory, Tsai and Hahn (1980),

$$\begin{Bmatrix} N \\ M \end{Bmatrix} = \begin{Bmatrix} A & B \\ B & D \end{Bmatrix} \begin{Bmatrix} \varepsilon^0 \\ \kappa \end{Bmatrix} \tag{2.1}$$

where N is a vector of resultant in-plane loads, M is a vector of resultant out-of-plane moments, ε^0 is the vector of mid-plane strains and κ is the vector of plate curvatures.

The in-plane, coupling and out-of-plane stiffness matrices are defined in terms of lamination parameters and material invariants,

$$\begin{pmatrix} A_{11} \\ A_{22} \\ A_{12} \\ A_{66} \\ A_{16} \\ A_{26} \end{pmatrix} = h \begin{bmatrix} 1 & \xi_1^A & \xi_2^A & 0 & 0 \\ 1 & -\xi_1^A & \xi_2^A & 0 & 0 \\ 0 & 0 & -\xi_2^A & 1 & 0 \\ 0 & 0 & -\xi_2^A & 0 & 1 \\ 0 & \xi_3^A/2 & \xi_4^A & 0 & 0 \\ 0 & \xi_3^A/2 & -\xi_4^A & 0 & 0 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{pmatrix} \quad \textbf{(2.2)}$$

$$\begin{pmatrix} B_{11} \\ B_{22} \\ B_{12} \\ B_{66} \\ B_{16} \\ B_{26} \end{pmatrix} = \frac{h^2}{4} \begin{bmatrix} 0 & \xi_1^B & \xi_2^B & 0 & 0 \\ 0 & -\xi_1^B & \xi_2^B & 0 & 0 \\ 0 & 0 & -\xi_2^B & 0 & 0 \\ 0 & 0 & -\xi_2^B & 0 & 0 \\ 0 & \xi_3^B/2 & \xi_4^B & 0 & 0 \\ 0 & \xi_3^B/2 & -\xi_4^B & 0 & 0 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{pmatrix} \quad \textbf{(2.3)}$$

$$\begin{pmatrix} D_{11} \\ D_{22} \\ D_{12} \\ D_{66} \\ D_{16} \\ D_{26} \end{pmatrix} = \frac{h^3}{12} \begin{bmatrix} 1 & \xi_1^D & \xi_2^D & 0 & 0 \\ 1 & -\xi_1^D & \xi_2^D & 0 & 0 \\ 0 & 0 & -\xi_2^D & 1 & 0 \\ 0 & 0 & -\xi_2^D & 0 & 1 \\ 0 & \xi_3^D/2 & \xi_4^D & 0 & 0 \\ 0 & \xi_3^D/2 & -\xi_4^D & 0 & 0 \end{bmatrix} \begin{pmatrix} U_1 \\ U_2 \\ U_3 \\ U_4 \\ U_5 \end{pmatrix} \quad \textbf{(2.4)}$$

where the lamination parameters are,

$$\begin{aligned} \xi^A_{[1,2,3,4]} &= \frac{1}{2}\int_{-1}^{1}[\cos 2\theta(z),\ \cos 4\theta(z),\ \sin 2\theta(z),\ \sin 4\theta(z)]dz \\ \xi^B_{[1,2,3,4]} &= \int_{-1}^{1}[\cos 2\theta(z),\ \cos 4\theta(z),\ \sin 2\theta(z),\ \sin 4\theta(z)]zdz \\ \xi^D_{[1,2,3,4]} &= \frac{3}{2}\int_{-1}^{1}[\cos 2\theta(z),\ \cos 4\theta(z),\ \sin 2\theta(z),\ \sin 4\theta(z)]z^2dz \end{aligned} \quad \textbf{(2.5)}$$

and $\theta(z)$ is the distribution function of the ply orientations through the normalised thickness co-ordinate $\bar{z} = \frac{2}{h}z$. For further details regarding $\theta(z)$, see Diaconu et al. (2002a). The material invariants are defined as,

$$
\begin{aligned}
U_1 &= [3Q_{11} + 3Q_{22} + 2Q_{12} + 4Q_{66}]/8 \\
U_2 &= [Q_{11} - Q_{12}]/2 \\
U_3 &= [Q_{11} + Q_{22} - 2Q_{12} - 4Q_{66}]/8 \\
U_4 &= [Q_{11} + Q_{22} + 6Q_{12} - 4Q_{66}]/8 \\
U_5 &= [Q_{11} + Q_{22} - 2Q_{12} + 4Q_{66}]/8
\end{aligned}
\qquad \textbf{(2.6)}
$$

where Q_{ij} are reduced stiffnesses for unidirectional lamina and defined as,

$$
\begin{aligned}
Q_{11} &= E_{11}^2/(E_{11} - E_{22}v_{12}^2) \\
Q_{22} &= E_{11}E_{22}/(E_{11} - E_{22}v_{12}^2) \\
Q_{12} &= v_{12}Q_{22} \\
Q_{66} &= G_{12}
\end{aligned}
\qquad \textbf{(2.7)}
$$

In Eqn. (2.7), E_{11}, E_{22}, G_{12} are the longitudinal, transverse, and shear moduli, v_{12} is the larger Poisson's ratio for a unidirectional laminate. Note, the 12 lamination parameters defined in Eqn. (2.5) are integrals through the thickness of the sines and cosines of the lay-up orientations. In practice, this integration is replaced by a through-the-thickness summation at ply level. Moreover, only these twelve lamination parameters are necessary to model the stiffness properties of any monolithic laminated composite.

The derivation of relationships between lamination parameters to provide a feasible region has been partially developed in an incremental fashion by several authors. The earliest use of lamination parameters in composite design appears to have been undertaken by Miki (1982) and Miki and Sugiyama (1993) who pioneered their use in optimization studies. In doing so, they defined the feasible region between some of the lamination parameters. Specifically, they derived (from first principles) the feasible regions needed to describe both the in-plane or out-of-plane stiffnesses of an orthotropic laminate using two in-plane or two out-of-plane lamination parameters respectively,

$$2(\xi_1^j)^2 - 1 \leq \xi_2^j \qquad \textbf{(2.8)}$$

where j = A, D. These feasible regions were used to provide an efficient, accurate and graphical approach to the design of laminated composites. Later, Fukunaga and Sekine (1992) derived the feasible regions of the four in-plane and separately, four out-of-plane, lamination parameters,

$$2(1+\xi_2^j)(\xi_3^j)^2 - 4\xi_1^j\xi_3^j\xi_4^j + (\xi_4^j)^2 - (\xi_2^j - 2(\xi_1^j)^2 + 1)(1-\xi_2^j) \leq 0 \qquad \textbf{(2.9)}$$

$$\left(\xi_1^j\right)^2 + \left(\xi_3^j\right)^2 \leq 1 \qquad \textbf{(2.10)}$$

where j = A, D.

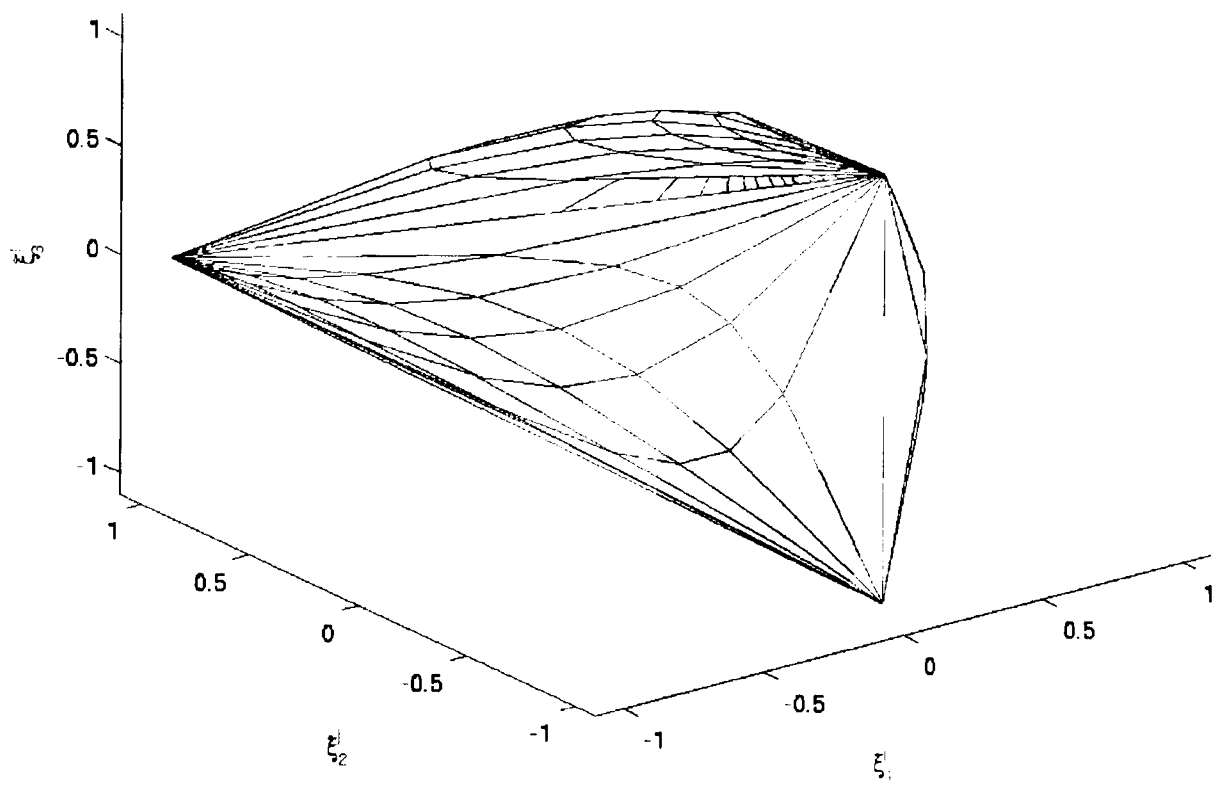

Fig. 2.1 – The Feasible Region for $\xi_1^j, \xi_2^j, \xi_3^j$ with $\xi_4^j = 0$ with Angles Unrestricted and where j = A,D

For a solely out-of-plane problem, such as initial buckling, knowledge of the complete feasible region makes the optimization process more efficient. At this time, the feasible region of the coupling lamination parameters, where ply orientations are unrestricted, has not been derived. Next, Grenestedt and Gudmundson (1993) used a variational approach to numerically determine the feasible region of orthotropic symmetric laminates. Furthermore, Grenestedt and Gudmundson (1993) derived explicit expressions between certain sets of in-plane and out-of-plane lamination parameters. For example,

$$\frac{1}{4}(\xi_i^A + 1)^3 - 1 \leq \xi_i^D \leq \frac{1}{4}(\xi_i^A - 1)^3 + 1 \qquad \textbf{(2.11)}$$

where i = 1...4. The feasible region represented by Eqn. (2.11) is shown in Fig. 2.2.

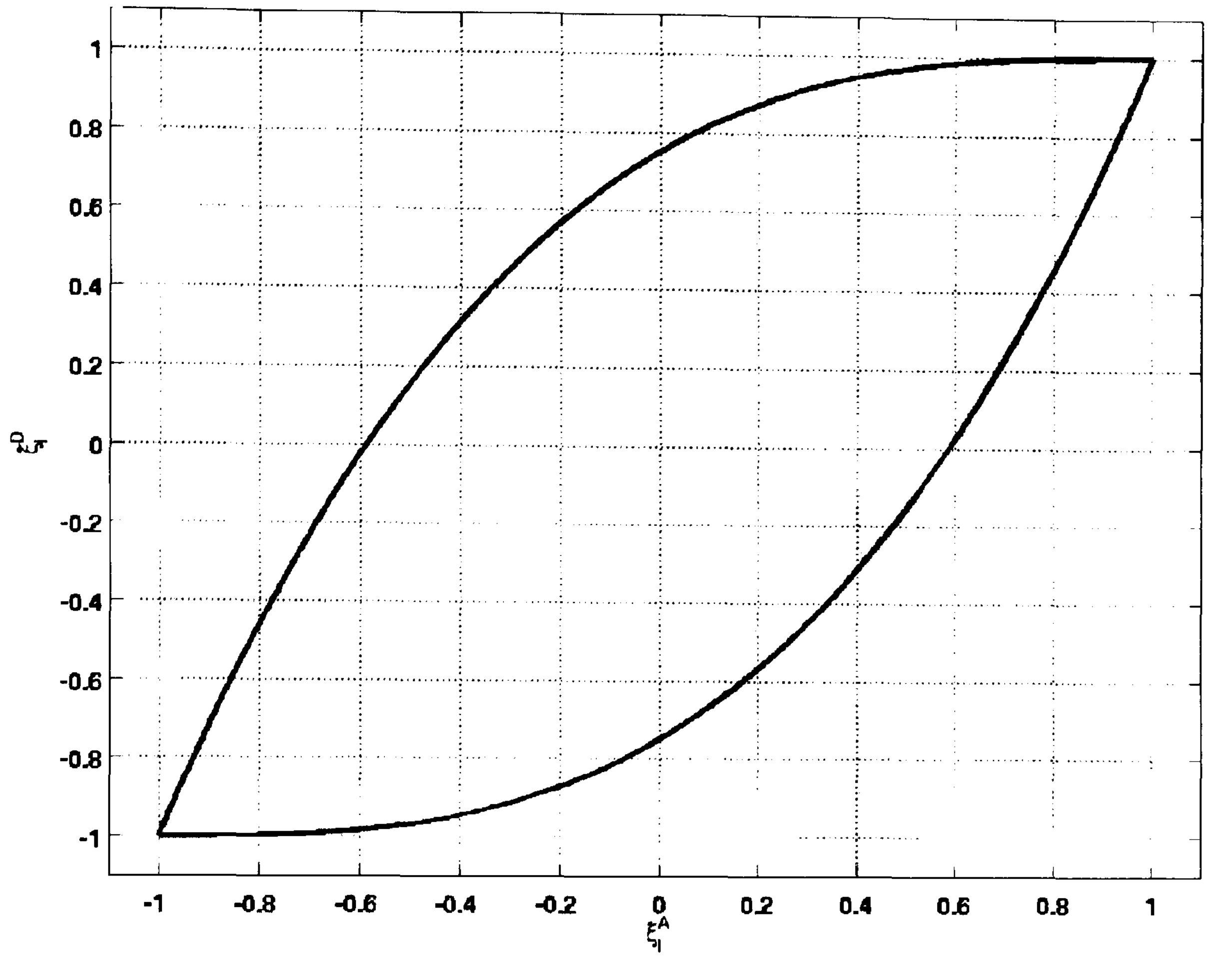

Fig. 2.2 – The Feasible Region of ξ_i^A, ξ_i^D for i = 1,2,3,4 with Angles Unrestricted

Note, for a given value of ξi_i^A, there exists a range of values for ξ_i^D for all values except for $\xi_i^A = \pm 1$ when $\xi_i^A = \xi_i^D$. Grenestedt and Gudmundson (1993) additionally proved that the feasible region was necessarily convex. Diaconu et al. (2002a) used the approach of Grenestedt and Gudmundson (1993) to obtain an implicit mathematical formulation of the general feasible region of all 12 lamination parameters. Also, Diaconu et al. (2002b) derived the feasible region of lamination parameters linking the in-plane, coupling and out-of-plane lamination parameters, where the index of the parameter was the same.

$$4(\xi_i^A + 1)(\xi_i^D + 1) \geq (\xi_i^A + 1)^4 + 3(\xi_i^B)^2 \qquad \textbf{(2.12)}$$

$$4(\xi_i^A - 1)(\xi_i^D - 1) \geq (\xi_i^A - 1)^4 + 3(\xi_i^B)^2 \qquad \textbf{(2.13)}$$

where i = 1...4. It is noted that in the above studies, no restrictions were placed on potential ply orientations. Later, Diaconu and Sekine (2004) derived explicitly the feasible regions of lamination parameters for $0, 90, \pm 45$ degree plies. They derived explicit expressions that related the in-plane, coupling and out-of-plane lamination parameters to each other. It is noted that Diaconu and Sekine (2004) did not provide a general method to determine the constraints on the feasible region for in-plane, coupling and out-of-plane lamination parameters for finite sets of ply orientations. One of the aims of the current Chapter is to provide such a method.

Todoroki and Terada (2004) presented an alternative approach to determine the feasible region of lamination parameters based upon a branch and bound approach, which is discussed in Chapter 4. Whilst the approach was successful, it is observed that Todoroki and Terada considered only the feasible regions of the in-plane and separately out-of-plane feasible regions. The 12 dimensional feasible region of the in-plane, coupling and out-of-plane lamination parameters inclusively, was not considered. Additionally, the computational cost in determining the feasible regions using the branch and bound method is greater than the approach presented herein. Recently, Setoodeh et al. (2006) developed a method to approximate the boundary of the general feasible region for lamination parameters. They generated lay-ups of varying thickness and ply orientations and calculated the corresponding lamination parameters for each lay-up. The convex hull of the set of lamination parameters was taken and noted to increase with growing number of different ply orientations. By monitoring convergence, they determined vectors of lamination parameters on the boundary of the feasible region. The convex hull was then used to determine 12 dimensional linear approximations of the feasible region of lamination parameters. Whilst it is noted this method could be used for a finite set of ply orientations, only approximations to the feasible region would be determined. In contrast, the method presented in this Chapter yields exact constraints on the feasible region for any finite set of ply orientations. Furthermore, the nature of the feasible region, as three interconnected discrete spaces, is not represented in the work of Setoodeh et al (2006). With respect to the method presented by Setoodeh et al. (2006), it is noted that this method yields a relatively large number of constraints, which may be computationally inefficient for optimization routines. It is further observed that computing the convex hull in higher dimensions is computationally expensive. Moreover, currently, industry generally restricts itself to discrete sets of ply orientations and, as such, the work presented herein may be useful for such purposes.

Additionally, it is observed that whilst the feasible region is determined for finite sets of ply orientations, ply thickness is assumed to be continuous. If ply thickness was assumed to be discrete, the resulting space would be a series of points in lamination parameter space. In this case, it would be difficult to use a continuous gradient based optimization.

At this time, the general feasible region remains unknown in analytical form. However, the method detailed in this chapter yields all the constraints on the feasible region of lamination parameters for any finite set of ply orientations. Before constructing new relationships between lamination parameters it is helpful to make some definitions.

Feasible Region – A region in an abstract design space of lamination parameters that contains all feasible vectors of lamination parameters. For each feasible vector of lamination parameters there exists at least one real lay-up. Note that for any given vector of lamination parameters outside the feasible region, no real lay-up exists. Also note that the dimension of the design space (or of the feasible region) equals the number of lamination parameters considered – maximum of 12 for the most general case.

Hyperplane Constraint – A linear inequality constraint forming a section the in-plane, coupling or out-of-plane feasible region denoted by H_A, H_B, H_D respectively.

Constraint on the Feasible Region - Any constraint which forms part of the boundary of the feasible region of lamination parameters

Using this background information and initial definitions, the method to derive the feasible region of the in-plane, coupling or out of-plane lamination parameters is presented in the following section.

2.3 Feasible Regions of In-Plane, Coupling and Out-of-Plane Lamination Parameters

In this section, the feasible region of the in-plane, coupling and out-of-plane lamination parameters is determined separately, using convex hulls. Formally, the convex hull of a finite set of points X is defined as,

$$C_H(X) = \left\{ \sum_{i=1}^{N} \lambda_i X_i \;\middle|\; \lambda_i \geq 0,\ i = 1 \ldots N,\ \sum_{i=1}^{N} \lambda_i = 1 \right\} \tag{2.14}$$

Note, the convex hull of a set of points, X is the minimum convex set containing X.

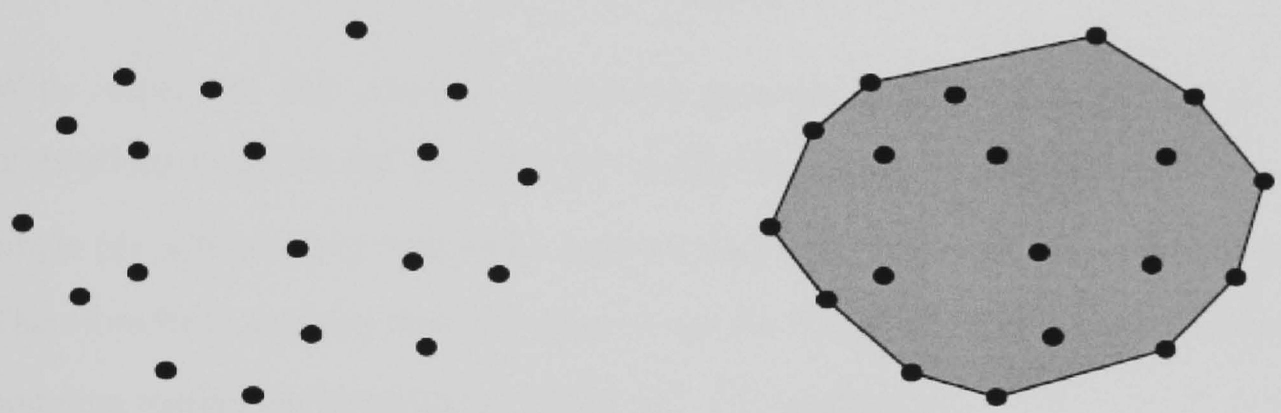

Fig. 2.3 – Graphical Definition of a 2D Convex Hull (Bertsekas et al. 2003)

For a finite set of ply orientations, it will be shown in this section that the feasible region of the in-plane, coupling or out-of-plane lamination parameters is a convex polyhedron (or polygon). Furthermore, the convex polyhedron is formed by taking the convex hull of the minimum number of vertices on the boundary of the feasible region. It is observed that a convex polyhedron is bounded by a set of hyperplanes. As such, the feasible region of the in-plane, coupling or out-of-plane lamination parameters is enclosed by a set of hyperplane constraints, each of dimension n where n is the number of the in-plane, coupling or out-of-plane feasible region and n = 1, 2, 3 or 4.

With respect to the in-plane or out-of-plane lamination parameters, the minimum number of plies which form the boundary of the feasible region is one, as shown by Fukunaga and Sekine (1992). Therefore, for any one ply of arbitrary orientation, the four in-plane or the four out-of-plane lamination parameters that form the boundary of the feasible region of the in-plane or out-of-plane lamination parameters, respectively, are defined as follows,

$$\begin{aligned} \xi_1^j &= \cos(2\theta) \\ \xi_2^j &= \cos(4\theta) \\ \xi_3^j &= \sin(2\theta) \\ \xi_4^j &= \sin(4\theta) \end{aligned} \qquad \text{where } j = A, D \tag{2.15}$$

With respect to the coupling lamination parameters, the minimum number of ply orientations to define the boundary of the feasible region is two (where $\theta_1 \neq \theta_2$) since a single ply is necessarily symmetric and two plies is the minimal needed for non-zero ξ_i^B. Therefore, it is asserted that two plies lie on the boundary of the feasible region of the coupling lamination parameters, $\xi_1^B, \xi_2^B, \xi_3^B, \xi_4^B$, defined as

$$\begin{aligned} \xi_1^B &= \frac{-\cos(2\theta_1) + \cos(2\theta_2)}{2} \\ \xi_2^B &= \frac{-\cos(4\theta_1) + \cos(4\theta_2)}{2} \\ \xi_3^B &= \frac{-\sin(2\theta_1) + \sin(2\theta_2)}{2} \\ \xi_4^B &= \frac{-\sin(4\theta_1) + \sin(4\theta_2)}{2} \end{aligned} \qquad \text{where } \theta_1 \neq \theta_2 \tag{2.16}$$

and θ_1 is the bottom ply. Equations (2.15) and (2.16) define vertices on the boundary of the feasible region of the in-plane, out-of-plane or coupling lamination parameters, respectively. Each vertex corresponds to a single ply orientation for in-plane or out-of-plane lamination parameters and to a non-symmetric combination of two plies of equal thickness and different angles for coupling lamination parameters. By connecting the vertices on the boundary of the feasible region, hyperplane constraints are determined and the explicit feasible regions derived. The set of vertices on the boundary of the feasible region are used in the following algorithm to analytically determine, separately, the feasible region of the in-plane, coupling and out-of-plane lamination parameters.

Algorithm 2.1

1) For each ply orientation or unique set of two orientations in ϕ, calculate the corresponding vertex of lamination parameters v, using Eqn. (2.15) or (2.16) respectively.
2) The set of all v for a given ϕ is denoted by V.
3) The convex hull of V is computed in MATLAB using QHULL (Barber 1996, Bertsekas et al. 2003) and the 'convhull' function (if $n = 2$) or 'convhulln function' (if $n > 2$).
4) The convex hull function outputs a series of sets of vertices. These sets of vertices are used to determine the coefficients of the hyperplane constraints which form the boundary of the feasible region of the in-plane, coupling or out-of-plane lamination parameters.

Note, if n =1, the range of ξ_k^j, with $j = A, D$ and k = 1...4 is calculated using Eqn. (2.15) for all $\theta \in \phi$. The range of the coupling lamination parameters is similarly calculated using Eqn. (2.16). For n =2, 3 or 4 the hyperplane constraints (which form the boundary of the in-plane, coupling and out-of-plane feasible regions) are found in the following form (n = 4 is shown):

$$\bar{h}_1^j \xi_1^j + \bar{h}_2^j \xi_2^j + \bar{h}_3^j \xi_3^j + \bar{h}_4^j \xi_4^j - \bar{h}_5^j = 0 \qquad \textbf{(2.17)}$$

where j = A, B, D and,

$$\bar{h}_1^j = \det\begin{pmatrix} 1 & \xi_{21}^j & \xi_{31}^j & \xi_{41}^j \\ 1 & \xi_{22}^j & \xi_{32}^j & \xi_{42}^j \\ 1 & \xi_{23}^j & \xi_{33}^j & \xi_{43}^j \\ 1 & \xi_{24}^j & \xi_{34}^j & \xi_{44}^j \end{pmatrix} \quad \bar{h}_2^j = \det\begin{pmatrix} \xi_{11}^j & 1 & \xi_{31}^j & \xi_{41}^j \\ \xi_{12}^j & 1 & \xi_{32}^j & \xi_{42}^j \\ \xi_{13}^j & 1 & \xi_{33}^j & \xi_{43}^j \\ \xi_{14}^j & 1 & \xi_{34}^j & \xi_{44}^j \end{pmatrix} \quad \bar{h}_3^j = \det\begin{pmatrix} \xi_{11}^j & \xi_{21}^j & 1 & \xi_{41}^j \\ \xi_{12}^j & \xi_{22}^j & 1 & \xi_{42}^j \\ \xi_{13}^j & \xi_{23}^j & 1 & \xi_{43}^j \\ \xi_{14}^j & \xi_{24}^j & 1 & \xi_{44}^j \end{pmatrix}$$

$$\bar{h}_4^j = \det\begin{pmatrix} \xi_{11}^j & \xi_{21}^j & \xi_{31}^j & 1 \\ \xi_{12}^j & \xi_{22}^j & \xi_{32}^j & 1 \\ \xi_{13}^j & \xi_{23}^j & \xi_{33}^j & 1 \\ \xi_{14}^j & \xi_{24}^j & \xi_{34}^j & 1 \end{pmatrix} \quad \bar{h}_5^j = -\det\begin{pmatrix} \xi_{11}^j & \xi_{21}^j & \xi_{31}^j & \xi_{41}^j \\ \xi_{12}^j & \xi_{22}^j & \xi_{32}^j & \xi_{42}^j \\ \xi_{13}^j & \xi_{23}^j & \xi_{33}^j & \xi_{43}^j \\ \xi_{14}^j & \xi_{24}^j & \xi_{34}^j & \xi_{44}^j \end{pmatrix} \tag{2.18}$$

In Eqn. (2.18), each entry in the determinants, ξ_{kl}^j, corresponds to a lamination parameter ξ_k^j with $j = A$, B and D and $k = 1..4$ for a specific vertex $l = 1..4$. Note that at least n vertices are necessary to define a hyperplane in an n-dimensional space. For simplicity, each hyperplane constraint is normalised with respect to the constant $\bar{h}_5^j$. Thus, Eqn. (2.17) can be rewritten as,

$$\frac{\bar{h}_1^j}{\bar{h}_5^j}\xi_1^j + \frac{\bar{h}_2^j}{\bar{h}_5^j}\xi_2^j + \frac{\bar{h}_3^j}{\bar{h}_5^j}\xi_3^j + \frac{\bar{h}_4^j}{\bar{h}_5^j}\xi_4^j - 1 = 0 \tag{2.19}$$

Substituting $h_i^j = \dfrac{\bar{h}_i^j}{\bar{h}_5^j}$ where $i = 1...4$, it follows that,

$$h_1^j\xi_1^j + h_2^j\xi_2^j + h_3^j\xi_3^j + h_4^j\xi_4^j - 1 = 0 \tag{2.20}$$

Note, $\bar{h}_5^j$ is a non-zero constant since the set of lamination parameters are linearly independent. Furthermore, if $\bar{h}_5^j$ was zero, the resulting hyperplane would pass through the origin in lamination parameter space. Since the origin is not on the boundary of the feasible region, $\bar{h}_5^j$ must always be non-zero. In order to validate Algorithm 2.1 and the assertions in the introduction of this section, the in-plane, coupling and out-of-plane feasible regions were derived for 0, 90, ± 45 degrees and proved to be identical with those derived by Diaconu and Sekine (2004). Moreover, using Algorithm 2.1, the explicit expressions describing the boundary of the aforementioned feasible regions for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies are derived and shown in Chapter 3. These feasible regions are shown in Figs. 2.4 and 2.5,

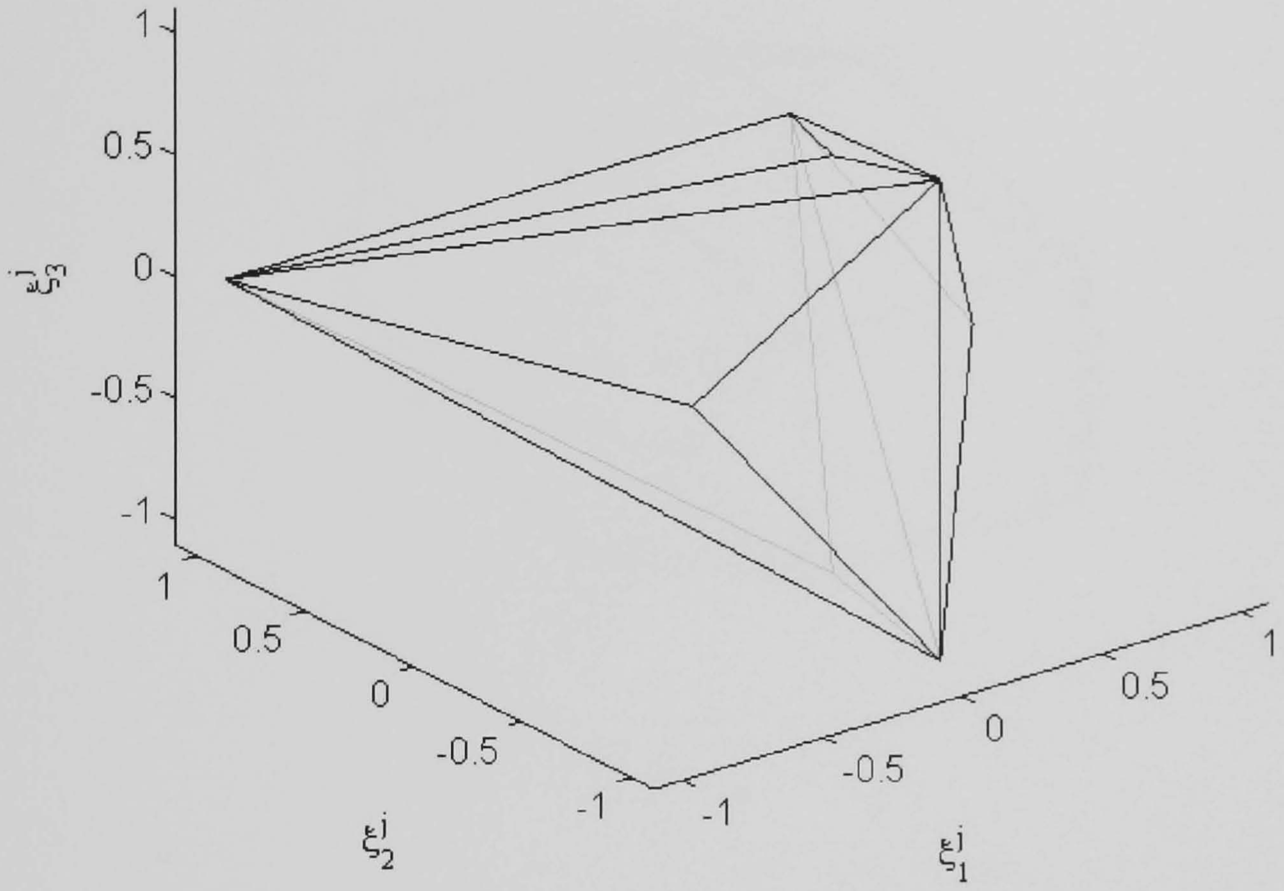

Fig. 2.4 - feasible region of $(\xi_1^j, \xi_2^j, \xi_3^j)$ **where** $\xi_4^j = 0$ **for** $0, 90, \pm 30, \pm 45, \pm 60$ **degree plies where** ***j*** **= A, D**

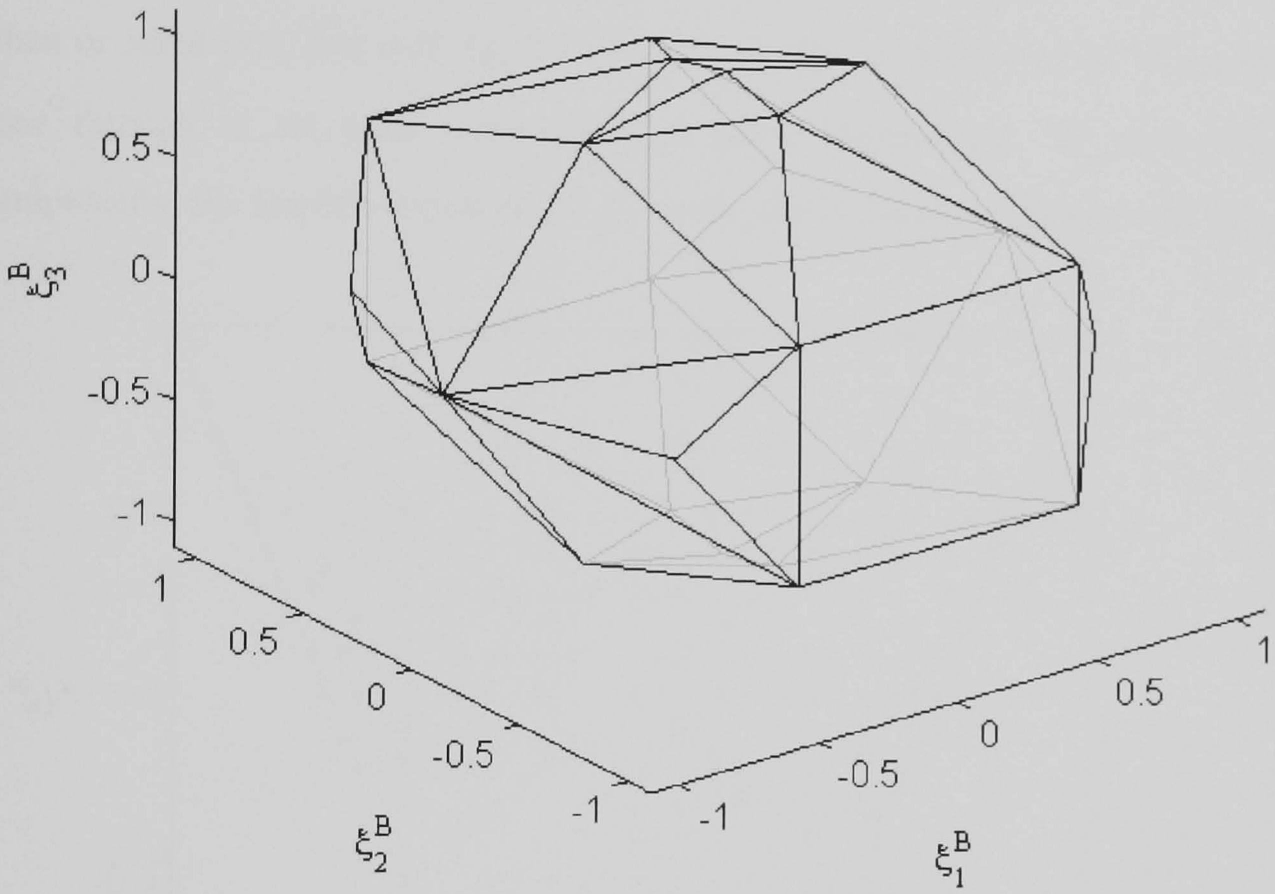

Fig. 2.5 - feasible region of $(\xi_1^B, \xi_2^B, \xi_3^B)$ **where** $\xi_4^B = 0$ **for** $0, 90, \pm 30, \pm 45, \pm 60$ **degree plies**

In Figs. 2.4 and 2.5 the light grey lines represent the boundary of planes on the hidden side of the surface. It is observed that the number of hyperplane constraints which enclose the feasible region of coupling lamination parameters is significantly greater than the number of in-plane or out-of-plane hyperplane constraints. Next, several tests are undertaken to validate that the obtained hyperplane constraints define the boundary of the feasible region of in-plane, coupling or out-of-plane lamination parameters. These tests show that the boundary of the feasible regions, obtained using Algorithm 2.1, is indeed the convex hull of the minimum number of vertices on the boundary of the feasible region.

The first test is undertaken to show that each feasible region is sufficiently large to include all possible laminate lay-ups from $0, 90, \pm 30, \pm 45, \pm 60$ degree plies. To demonstrate this feature, each vector of feasible lamination parameters ξ, when substituted into each hyperplane constraint, denoted H_j where $j = A, B, D$, must be less than or equal to 0, that is $H_j(\xi) \leq 0$. To achieve this all combinations of angles in ϕ, of one through to six plies were generated using enumeration. To show this feature graphically, the feasible region of ξ_1^A, ξ_2^A was selected and is shown in Fig. 2.6.

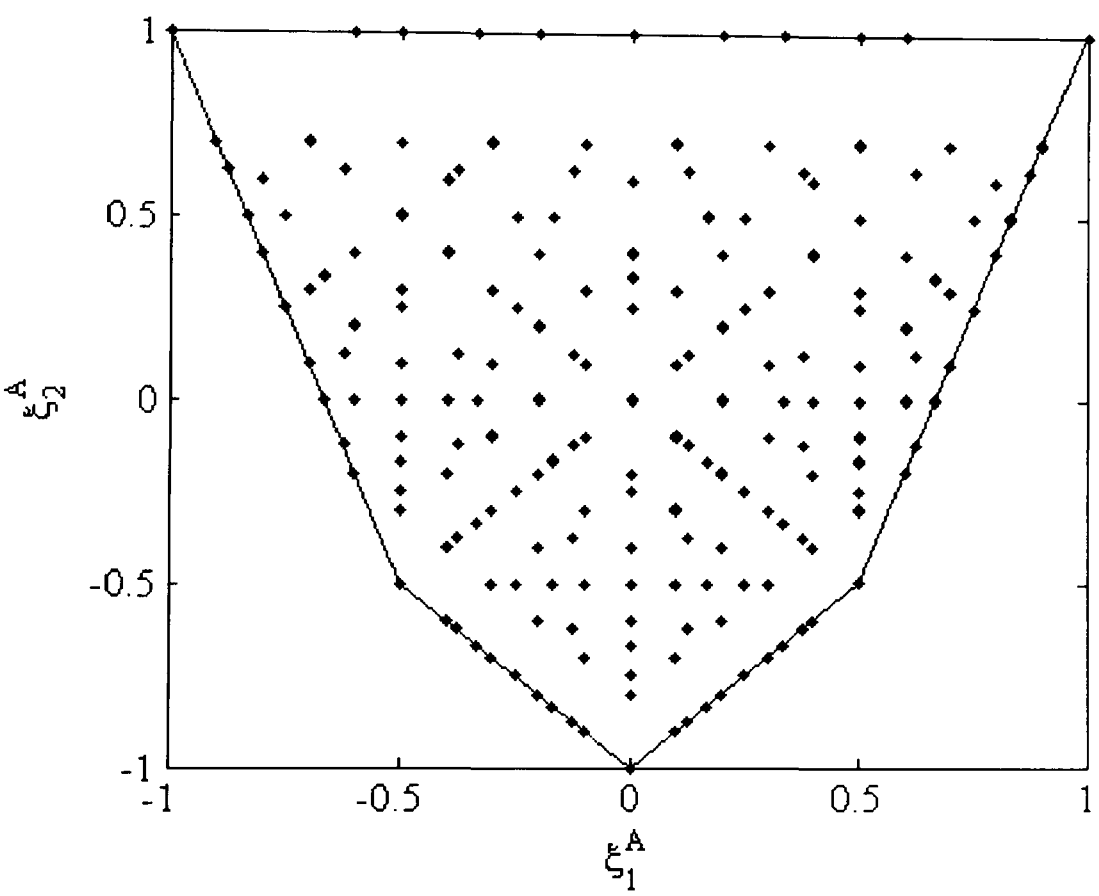

Fig. 2.6 – feasible vectors of (ξ_1^A, ξ_2^A) for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies assuming uniform ply thickness

A second test is undertaken to determine whether each feasible region is sufficiently small to exclude potentially infeasible sets of lamination parameters. For every vector of lamination parameters ξ that satisfies the set of hyperplane constraints, a real lay-up is sought: if the Euclidean distance between ξ and the vector of lamination parameters corresponding to the determined lay-up is small, then the set of lamination parameters has passed the test. A discrete optimizer (see Chapters 4 and 5) was used to determine a lay-up from a given ξ. These two tests confirm the extent of the derived feasible region and, moreover, highlight the fact, that the feasible region is formed by the convex hull of the minimum set of vertices on its boundary.

2.4 Determining the Feasible Regions of Lamination Parameters

In this section, the second level necessary to determine the feasible regions of lamination parameters is presented. Specifically, a method for establishing relationships between the constraints on the in-plane, coupling and out-of-plane feasible regions (as found in the previous section) is derived. Note, the method builds upon the work of Diaconu and Sekine (2004) and generalises their approach such that it is valid for any finite set of ply orientations.

Firstly, the in-plane lamination parameters defined in Eqn. (2.5), for a lay-up of p plies are rewritten as a finite sum, thus,

$$\begin{aligned}
\xi_1^A &= \frac{1}{2}\left(c_2(1) + z_1(c_2(1) - c_2(2))\ldots + z_{p-1}(c_2(p-1) - c_2(p)) + c_2(p)\right) \\
\xi_2^A &= \frac{1}{2}\left(c_4(1) + z_1(c_4(1) - c_4(2))\ldots + z_{p-1}(c_4(p-1) - c_4(p)) + c_4(p)\right) \\
\xi_3^A &= \frac{1}{2}\left(s_2(1) + z_1(s_2(1) - s_2(2))\ldots + z_{p-1}(s_2(p-1) - s_2(p)) + s_2(p)\right) \\
\xi_4^A &= \frac{1}{2}\left(s_4(1) + z_1(s_4(1) - s_4(2))\ldots + z_{p-1}(s_4(p-1) - s_4(p)) + s_4(p)\right)
\end{aligned} \qquad \textbf{(2.21)}$$

where,

$$\begin{aligned}
c_2(i) &= \cos 2\theta_i \\
c_4(i) &= \cos 4\theta_i \\
s_2(i) &= \sin 2\theta_i \\
s_4(i) &= \sin 4\theta_i
\end{aligned} \qquad \textbf{(2.22)}$$

and z_i $(i = 1 \ldots p\text{-}1)$ is the normalised through thickness co-ordinate of each ply. Note that $z_0 = -1$ and $z_p = 1$. Coupling and out-of-plane lamination parameters are similarly defined by making the appropriate square and cubic substitutions for z_i in Eqn. (2.21). Next, explicit expressions linking lamination parameters from each design subspace are found. Motivated by the algebraic identity, first used by Diaconu and Sekine (2002b)

$$4(z_x - z_y)(z_x^3 - z_y^3) = (z_x - z_y)^4 + 3(z_x^2 - z_y^2)^2 \qquad \textbf{(2.23)}$$

it is later proven that on the boundary of the feasible region,

$$\begin{aligned} \frac{1}{k}\left(h_1\xi_1^A + h_2\xi_2^A + h_3\xi_3^A + h_4\xi_4^A + h_5\right) &= z_x - z_y \\ \frac{1}{k}\left(h_1\xi_1^B + h_2\xi_2^B + h_3\xi_3^B + h_4\xi_4^B\right) &= z_x^2 - z_y^2 \\ \frac{1}{k}\left(h_1\xi_1^D + h_2\xi_2^D + h_3\xi_3^D + h_4\xi_4^D + h_5\right) &= z_x^3 - z_y^3 \end{aligned} \qquad \textbf{(2.24)}$$

where,

$$k = \max\left(\frac{1}{2}\sum_{i=1}^{5} h_i\,\xi_i^A\right) \qquad \textbf{(2.25)}$$

and x, y in Eqns. (2.23-2.24) are integers, where $x, y \in [1, p-1]$, $\xi_5^A = 1$ and $x \neq y$, Eqn. (2.23) has fundamental importance for establishing relationships between lamination parameters since it makes links between the linear, quadratic and cubic volume fractions of each ply orientation in the lay-up and thus the *A, B, D* stiffness matrices. To make these connections, Eqns. (2.24) are substituted into Eqn (2.23) and the resulting expression is multiplied through by k^4. to obtain

$$4k^2\left(\sum_{i=1}^{5} h_i\xi_i^A\right)\left(\sum_{i=1}^{5} h_i\xi_i^D\right) \geq \left(\sum_{i=1}^{5} h_i\xi_i^A\right)^4 + 3k^2\left(\sum_{i=1}^{4} h_i\xi_i^B\right)^2 \quad . \qquad \textbf{(2.26)}$$

where $\xi_5^j = 1$ and j = *A*, *B*, *D*, noting that the inequality represents the scope of the feasible region. Furthermore, Eqn. (2.26) is the fundamental explicit expression that establishes relationships between in-plane, out-of-plane and coupling hyperplane constraints and thus lamination parameters to each other. Note, in Eqn. (2.26), the inequality has been introduced to show that the constraint forms a section on the boundary of the feasible region of lamination parameters. For lamination parameters on the boundary of the feasible region, the inequality in Eqn. (2.26) becomes a strict equality. It is noted from Diaconu et al. (2004) that for each constraint in the form of Eqn. (2.26), there were two values for h_5. The two values of h_5 arise from the existence of an upper and lower bound value for each hyperplane constraint (which was determined at the first level). This assertion can be proved as follows. Since each lamination parameter is a simple trigonometric function, its value is bounded by ± 1. It follows that any linear combination of lamination parameters has a lower and upper bound. Therefore each hyperplane constraint has a lower and an upper bound, denoted by H^L and H^U, respectively. Specifically, the values of H^L and H^U are defined as the minimum and maximum values of $\sum_{i=1}^{4} h_i \xi_i^A$. Substituting H^L, H^U for h_5 in Eqn. (2.26) gives,

$$4k^2\left(\sum_{i=1}^{4} h_i \xi_i^A - H^L\right)\left(\sum_{i=1}^{4} h_i \xi_i^D - H^L\right) \geq \left(\sum_{i=1}^{4} h_i \xi_i^A - H^L\right)^4 + 3k^2\left(\sum_{i=1}^{4} h_i \xi_i^B\right)^2 \quad \textbf{(2.27)}$$

and

$$4k^2\left(\sum_{i=1}^{4} h_i \xi_i^A - H^U\right)\left(\sum_{i=1}^{4} h_i \xi_i^D - H^U\right) \geq \left(\sum_{i=1}^{4} h_i \xi_i^A - H^U\right)^4 + 3k^2\left(\sum_{i=1}^{4} h_i \xi_i^B\right)^2. \quad \textbf{(2.28)}$$

Theorem

The constraints on the feasible region of the lamination parameters are interrelated and can be expressed in the form expressed in Eqns. (2.27-2.28).

Proof

To prove the above statement, it is asserted that the hyperplane constraints, $h_i\xi_i^j$, where j $=A, B, D$, are related to each other by the following algebraic identity,

$$4(z_x - z_y)(z_x^3 - z_y^3) = (z_x - z_y)^4 + 3(z_x^2 - z_y^2)^2 \quad \textbf{(2.29)}$$

where

$$\begin{aligned} \frac{1}{k}\left(h_1\xi_1^A + h_2\xi_2^A + h_3\xi_3^A + h_4\xi_4^A + h_5\right) &= z_x - z_y \\ \frac{1}{k}\left(h_1\xi_1^B + h_2\xi_2^B + h_3\xi_3^B + h_4\xi_4^B\right) &= z_x^2 - z_y^2 \end{aligned} \quad \textbf{(2.30)}$$

$$\frac{1}{k}\left(h_1\xi_1^D + h_2\xi_2^D + h_3\xi_3^D + h_4\xi_4^D + h_5\right) = z_x^3 - z_y^3$$

Note, k is a scaling factor and x, y are integers and $\in [1, p-1]$. A proof follows.

Firstly, we seek to establish the existence of an identity in the form of Eqn. (2.29). There is existing evidence to suggest that the form of Eqn (2.29) is appropriate because it was demonstrated by Diaconu and Sekine (2002b) that

$$4(\xi_1^A + 1)(\xi_1^D + 1) = (\xi_1^A + 1)^4 + 3(\xi_1^B)^2 \qquad \textbf{(2.31)}$$

which simplifies to Eqn. (2.12) when $h_1 = 1, h_2, h_3, h_4 = 0, h_5 = 1$ and $k = 1$ are substituted into Eqn. (2.30). Therefore, one solution is known to exist in the form of Eqn. (2.31).

To proceed further, the identity of Eqn. (2.31) is generalised to fit the form of Eqns. (2.29-2.30). A solution is assumed to exist in the following form,

$$\begin{aligned}
\alpha_1\xi_1^A + \alpha_2\xi_2^A + \alpha_3\xi_3^A + \alpha_4\xi_4^A + \alpha_5 &= z_x - z_y \\
\beta_1\xi_1^B + \beta_2\xi_2^B + \beta_3\xi_3^B + \beta_4\xi_4^B + \beta_5 &= z_x^2 - z_y^2 \\
\chi_1\xi_1^D + \chi_2\xi_2^D + \chi_3\xi_3^D + \chi_4\xi_4^D + \chi_5 &= z_x^3 - z_y^3
\end{aligned} \qquad \textbf{(2.32)}$$

where $\alpha_i, \beta_i, \chi_i$ are real valued scalars. Using Eqn. (2.5), the in-plane lamination parameters can be re-written as

$$\begin{aligned}
\xi_1^A &= \frac{1}{2}\left(c_2(1) + z_1(c_2(1) - c_2(2))... + z_{p-1}(c_2(p-1) - c_2(p)) + c_2(p)\right) \\
\xi_2^A &= \frac{1}{2}\left(c_4(1) + z_1(c_4(1) - c_4(2))... + z_{p-1}(c_4(p-1) - c_4(p)) + c_4(p)\right) \\
\xi_3^A &= \frac{1}{2}\left(s_2(1) + z_1(s_2(1) - s_2(2))... + z_{p-1}(s_2(p-1) - s_2(p)) + s_2(p)\right) \\
\xi_4^A &= \frac{1}{2}\left(s4(1) + z_1(s_4(1) - s_4(2))... + z_{p-1}(s_4(p-1) - s_4(p)) + s_4(p)\right)
\end{aligned} \qquad \textbf{(2.33)}$$

where p is the number of plies in the lay-up. Note, coupling and out-of-plane lamination parameters can be similarly defined. Forming $\alpha_i\xi_i^A$, $\beta_i\xi_i^B$, $\chi_i\xi_i^D$ then,

$$\begin{aligned}
\alpha_1\xi_1^A &= \frac{\alpha_1}{2}\left(c_2(1) + z_1(c_2(1) - c_2(2))\ldots + z_{p-1}(c_2(p-1) - c_2(p)) + c_2(p)\right) \\
\alpha_2\xi_2^A &= \frac{\alpha_2}{2}\left(c_4(1) + z_1(c_4(1) - c_4(2))\ldots + z_{p-1}(c_4(p-1) - c_4(p)) + c_4(p)\right) \\
\alpha_3\xi_3^A &= \frac{\alpha_3}{2}\left(s_2(1) + z_1(s_2(1) - s_2(2))\ldots + z_{p-1}(s_2(p-1) - s_2(p)) + s_2(p)\right) \\
\alpha_4\xi_4^A &= \frac{\alpha_4}{2}\left(s_4(1) + z_1(s_4(1) - s_4(2))\ldots + z_{p-1}(s_4(p-1) - s_4(p)) + s_4(p)\right)
\end{aligned} \tag{2.34}$$

$$\begin{aligned}
\beta_1\xi_1^B &= \frac{\beta_1}{2}\left(-c_2(1) + z_1^2(c_2(1) - c_2(2))\ldots + z_{p-1}^2(c_2(p-1) - c_2(p)) + c_2(p)\right) \\
\beta_2\xi_2^B &= \frac{\beta_2}{2}\left(-c_4(1) + z_1^2(c_4(1) - c_4(2))\ldots + z_{p-1}^2(c_4(p-1) - c_4(p)) + c_4(p)\right) \\
\beta_3\xi_3^B &= \frac{\beta_3}{2}\left(-s_2(1) + z_1^2(s_2(1) - s_2(2))\ldots + z_{p-1}^2(s_2(p-1) - s_2(p)) + s_2(p)\right) \\
\beta_4\xi_4^B &= \frac{\beta_4}{2}\left(-s_4(1) + z_1^2(s_4(1) - s_4(2))\ldots + z_{p-1}^2(s_4(p-1) - s_4(p)) + s_4(p)\right)
\end{aligned} \tag{2.35}$$

$$\begin{aligned}
\chi_1\xi_1^D &= \frac{\chi_1}{2}\left(c_2(1) + z_1^3(c_2(1) - c_2(2))\ldots + z_{p-1}^3(c_2(p-1) - c_2(p)) + c_2(p)\right) \\
\chi_2\xi_2^D &= \frac{\chi_2}{2}\left(c_4(1) + z_1^3(c_4(1) - c_4(2))\ldots + z_{p-1}^3(c_4(p-1) - c_4(p)) + c_4(p)\right) \\
\chi_3\xi_3^D &= \frac{\chi_3}{2}\left(s_2(1) + z_1^3(s_2(1) - s_2(2))\ldots + z_{p-1}^3(s_2(p-1) - s_2(p)) + s_2(p)\right) \\
\chi_4\xi_4^D &= \frac{\chi_4}{2}\left(s_4(1) + z_1^3(s_4(1) - s_4(2))\ldots + z_{p-1}^3(s_4(p-1) - s_4(p)) + s_4(p)\right)
\end{aligned} \tag{2.36}$$

Summing components of Eqn. (2.34) gives

$$\begin{aligned}
\alpha_1\xi_1^A + \alpha_2\xi_2^A + \alpha_3\xi_3^A + \alpha_4\xi_4^A &= \frac{\alpha_1}{2}(c_2(1) + c_2(p) + \ldots) + \frac{\alpha_2}{2}(c_4(1) + c_4(p) + \ldots) \\
&\quad + \frac{\alpha_3}{2}(s_2(1) + s_2(p) + \ldots) + \frac{\alpha_4}{2}(s_4(1) + s_4(p) + \ldots)
\end{aligned} \tag{2.37}$$

By collecting the constant terms from Eqn. (2.37) and substituting them into Eqn. (2.32a), it follows that,

$$\alpha_5 = -\begin{pmatrix} \frac{\alpha_1}{2}(c_2(1)+c_2(p)) + \frac{\alpha_2}{2}(c_4(1)+c_4(p)) \\ +\frac{\alpha_3}{2}(s_2(1)+s_2(p)) + \frac{\alpha_4}{2}(s_4(1)+s_4(p)) \end{pmatrix} \tag{2.38}$$

Now define a vector Γ where,

$$\Gamma = \begin{matrix} \frac{1}{2}(\cos(2\theta_1)+\cos(2\theta_p)) \\ \frac{1}{2}(\cos(4\theta_1)+\cos(4\theta_p)) \\ \frac{1}{2}(\sin(2\theta_1)+\sin(2\theta_p)) \\ \frac{1}{2}(\sin(4\theta_1)+\sin(4\theta_p)) \end{matrix} \tag{2.39}$$

It follows from Eqn. (2.38) that,

$$\alpha_5 = -(\alpha_1 \; \alpha_2 \; \alpha_3 \; \alpha_4)\Gamma \quad . \tag{2.40}$$

It can be shown from Eqn. (2.17) that for a given hyperplane,

$$h_5 = -(h_1 \; h_2 \; h_3 \; h_4)\Gamma \quad . \tag{2.41}$$

when $\theta_1 = \theta_p$. Therefore, $\vec{\alpha} = \{\alpha_1 \quad \alpha_2 \quad \alpha_3 \quad \alpha_4\}^T$ must be parallel to $\vec{h} = \{h_1 \quad h_2 \quad h_3 \quad h_4\}^T$. Furthermore, since $\theta_1 = \theta_p$ it is evident that $\beta_5 = 0$. Similarly, it can be shown that $\vec{\beta} = \{\beta_1 \quad \beta_2 \quad \beta_3 \quad \beta_4\}^T$ is parallel to $\vec{h}$ and $\vec{\chi} = \{\chi_1 \quad \chi_2 \quad \chi_3 \quad \chi_4\}^T$ is parallel to $\vec{h}$. Therefore, by definition of parallelism of vectors

$$\vec{\alpha} = \psi_1 \vec{h} \qquad \vec{\beta} = \psi_2 \vec{h} \qquad \vec{\chi} = \psi_3 \vec{h} \tag{2.42}$$

where ψ_1, ψ_2, ψ_3 are scalars. Next., substituting Eqns. (2.42) into Eqns. (2.34), (2.35) and (2.32) respectively and making the appropriate summations,

$$\begin{aligned} \psi_1 h_1 \xi_1^A + \psi_1 h_2 \xi_2^A + \psi_1 h_3 \xi_3^A + \psi_1 h_4 \xi_4^A + \psi_1 h_5 &= z_x - z_y \\ \psi_2 h_1 \xi_1^B + \psi_2 h_2 \xi_2^B + \psi_2 h_3 \xi_3^B + \psi_2 h_4 \xi_4^B &= z_x^2 - z_y^2 \end{aligned} \tag{2.43}$$

$$\psi_3 h_1 \xi_1^D + \psi_3 h_2 \xi_2^D + \psi_3 h_3 \xi_3^D + \psi_3 h_4 \xi_4^D + \psi_3 h_5 = z_x^3 - z_y^3$$

and rearranging gives,

$$\begin{aligned} h_1 \xi_1^A + h_2 \xi_2^A + h_3 \xi_3^A + h_4 \xi_4^A + h_5 &= \frac{1}{\psi_1}(z_x - z_y) \\ h_1 \xi_1^B + h_2 \xi_2^B + h_3 \xi_3^B + h_4 \xi_4^B &= \frac{1}{\psi_2}(z_x^2 - z_y^2) \\ h_1 \xi_1^D + h_2 \xi_2^D + h_3 \xi_3^D + h_4 \xi_4^D + h_5 &= \frac{1}{\psi_3}(z_x^3 - z_y^3) \end{aligned} \quad \textbf{(2.44)}$$

By fully expanding Eqn. (2.37) in terms of z_i, and then combining with Eqn. (2.44) the coefficient of z_i for $i = 1 \ldots p\text{-}1$ is,

$$\frac{h_1}{2}(c_2(i) - c_2(i+1)) + \frac{h_2}{2}(c_4(i) - c_4(i+1)) + \frac{h_3}{2}(s_2(i) - s_2(i+1)) + \frac{h_4}{2}(s_4(i) - s_4(i+1)).$$

(2.45)

It follows from Eqns. (2.44-2.45) that,

$$\frac{1}{\psi_1} = \left(\frac{h_1}{2}(c_2(i) - c_2(i+1)) + \frac{h_2}{2}(c_4(i) - c_4(i+1)) + \frac{h_3}{2}(s_2(i) - s_2(i+1)) + \frac{h_4}{2}(s_4(i) - s_4(i+1))\right)$$

(2.46)

Similarly,

$$\frac{1}{\psi_2} = \left(\frac{h_1}{2}(c_2(i) - c_2(i+1)) + \frac{h_2}{2}(c_4(i) - c_4(i+1)) + \frac{h_3}{2}(s_2(i) - s_2(i+1)) + \frac{h_4}{2}(s_4(i) - s_4(i+1))\right)$$

(2.47)

$$\frac{1}{\psi_3} = \left(\frac{h_1}{2}(c_2(i) - c_2(i+1)) + \frac{h_2}{2}(c_4(i) - c_4(i+1)) + \frac{h_3}{2}(s_2(i) - s_2(i+1)) + \frac{h_4}{2}(s_4(i) - s_4(i+1))\right)$$

(2.48)

Comparing coefficients in Eqns. (2.46-2.48), it immediately follows $\psi_1 = \psi_2 = \psi_3$, which is denoted by $\frac{1}{k}$ herein and hence,

$$\frac{1}{k}\left(h_1 \xi_1^A + h_2 \xi_2^A + h_3 \xi_3^A + h_4 \xi_4^A + h_5\right) = z_x - z_y$$

$$\frac{1}{k}\left(h_1\xi_1^B + h_2\xi_2^B + h_3\xi_3^B + h_4\xi_4^B\right) = z_x^2 - z_y^2$$
$$\frac{1}{k}\left(h_1\xi_1^D + h_2\xi_2^D + h_3\xi_3^D + h_4\xi_4^D + \chi_5\right) = z_x^3 - z_y^3 \tag{2.49}$$

Therefore the general form of Eqn (2.29) is now proven. To simplify the expressions, we substitute Eqns. (2.49) into Eqn. (2.29) and multiplying through by k^4 yields,

$$\begin{aligned}&4k^2(h_1\xi_1^A + h_2\xi_2^A + h_3\xi_3^A + h_4\xi_4^A + h_5)(h_1\xi_1^D + h_2\xi_2^D + h_3\xi_3^D + h_4\xi_4^D + h_5) = \\ &(h_1\xi_1^A + h_2\xi_2^A + h_3\xi_3^A + h_4\xi_4^A + h_5)^4 + 3k^2(h_1\xi_1^B + h_2\xi_2^B + h_3\xi_3^B + h_4\xi_4^B)^2\end{aligned} \tag{2.50}$$

Solving Eqn. (2.50) for k^2 gives,

$$k^2 = \frac{(\sum_{i=1}^{5} h_i\xi_i^A)^4}{4(\sum_{i=1}^{5} h_i\xi_i^A)(\sum_{i=1}^{5} h_i\xi_i^D) - 3(\sum_{i=1}^{4} h_i\xi_i^B)^2} \tag{2.51}$$

where $\xi_5^{A,D} = 1$. Clearly, k depends upon the values of the in-plane, coupling and out-of-plane lamination parameters. Moreover, for each feasible lay-up, there is a corresponding value of k. The objective is then to find maximum k denoted $k_{\max}$ such that Eqn. (2.50) is convex and all feasible lay-ups are either on or within the boundary of the feasible region. The appropriate scaling factor, $k_{\max}$ is determined as follows,

$$k_{\max}^2 = \frac{\max\left\{\left|(\sum_{i=1}^{5} h_i\xi_i^A)^4\right|\right\}}{\max\left\{4(\sum_{i=1}^{5} h_i\xi_i^A)(\sum_{i=1}^{5} h_i\xi_i^D) - 3(\sum_{i=1}^{4} h_i\xi_i^B)^2\right\}} \tag{2.52}$$

It is observed that, $\max\left\{\left|(\sum_{i=0}^{5} h_i\xi_i^A)\right|\right\}$ is reached at a vertex which corresponds to one ply angle. This observation holds because in a convex polyhedron the maximum distance (positions at extreme points) from any bounding hyperplane is obtained at a vertex on the polyhedron. Moreover, this occurs at a vertex on the bounding hyperplane. Therefore the maximum distance occurs between two vertices. For details, see Bertsekas et al (2003).

Since one ply is necessarily symmetric, $\sum_{i=1}^{4} h_i \xi_i^B = 0$. Noting that $\max\left\{(\sum_{i=1}^{5} h_i \xi_i^A)\right\} = \max\left\{(\sum_{i=1}^{5} h_i \xi_i^D)\right\}$ and using the following identity

$$\max\left\{(\sum_{i=1}^{5} h_i \xi_i^A)^4\right\} = \left(\max\left\{\left|\sum_{i=1}^{5} h_i \xi_i^A\right|\right\}\right)^4 \quad \textbf{(2.53)}$$

it follows that,

$$k_{\max}{}^2 = \frac{\max\left((\sum_{i=1}^{5} h_i \xi_i^A)^4\right)}{\max\left(4(\sum_{i=1}^{5} h_i \xi_i^A)^2\right)} = \max\left(\frac{(\sum_{i=1}^{5} h_i \xi_i^A)^2}{4}\right) = \left(\max\left(\frac{\sum_{i=1}^{5} h_i \xi_i^A}{2}\right)\right)^2 \quad \textbf{(2.54)}$$

■

Eqns. (2.27-2.28) are the expressions that connect in-plane, coupling and out-of-plane hyperplane constraints, determined using Algorithm 2.1, to each other. It was observed in Section 3 that the number of constraints of the feasible region of coupling lamination parameters was significantly greater than the number of constraints on the in-plane or out-of-plane feasible regions. From Eqns. (2.27-2.28), it follows that for each constraint on the coupling lamination parameters, there must be a corresponding constraint on the in-plane and out-of-plane lamination parameters when all lamination parameters are explicitly related to one another. Moreover, when the constraints become equalities, $h_i = h_i^A = h_i^B = h_i^D$ where i = 1...4. Therefore, the feasible region of the coupling lamination parameters is used to derive the constraints on the entire 12 dimensional feasible region. The process of deriving the explicit expressions that make links between the lamination parameters is summarised in Algorithm 2.2.

Algorithm 2.2

1) Using Algorithm 2.1, determine, separately, the hyperplane constraints on the in-plane, coupling and out-of-plane feasible regions

2) For each hyperplane constraint on the coupling lamination parameters (h_i) found in Step 1, determine H^L, H^U using the minimum and maximum of $\sum_{i=1}^{4} h_i \xi_i^A$ and determine k using Eqn. (2.25).

3) Substitute H^L, H^U and k into Eqns. (2.27-2.28) to establish linking expressions between in-plane, coupling and out-of-plane feasible regions.

Using Algorithm 2.2, the explicit expressions that connect the in-plane, out-of-plane and coupling lamination parameters for the predefined finite set of $0, 90, \pm 30, \pm 45, \pm 60$ degree plies are detailed in Chapter 3 within the context of the optimization formulation. Next, the obtained constraints are tested and validated.

2.5 Validation and Confirmation of Results

In order to confirm that the set of derived constraints formed using Eqns. (2.27-2.28) fully define the feasible region of lamination parameters for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies, two tests are carried out. With respect to the first test, all feasible lay-ups should strictly obey the inequality constraint, e.g. all feasible lay-ups must lie on the boundary of the feasible region or inside of it. This test was detailed in section 2.3. Concerning the second test, each vertex on the boundary of the feasible region should correspond to at least one real lay-up. The second test can be summarised in three levels,

1) A random vector, ξ, of 12 lamination parameters is generated.
2) If ξ satisfies the set of constraints, continue to Step 3, otherwise return to Step 1.
3) For each ξ which satisfies the constraints, a corresponding lay-up is determined using a discrete optimizer (see Chapters 4 and 5).
4) The Euclidean distance between ξ and the corresponding vector of lamination parameters (calculated from the obtained lay-up in Step 3) is calculated. If the distance between the two vectors is small, then it is accepted that the test has been passed.

All lay-ups passed both tests which confirms that the derived feasible region for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies is indeed appropriate.

2.6 Elastic Tailoring Using the Design Space of Lamination Parameters

As sets of ply orientations correspond to feasible regions of lamination parameters, it follows that certain sets of ply orientations yield ranges of values for the stiffness matrices. The following aims to identify the relationship between various sets of ply orientations and the size of the feasible region it occupies. It is expected that such information will be useful for the elastic tailoring of composite laminates. Numerical examples will confirm this in Chapter 7.

Lamination parameters, due to their simple trigonometric nature, are bounded between non-dimensional values of -1 and 1. Each four dimensional space, therefore has a hyperspace volume of 2^4 which is 16. However, the actual design space is significantly smaller due to the limiting constraining relationships between lamination parameters. Using the method detailed in this Chapter, it is possible to evaluate the volume of design space for different sets of allowable ply orientations. The expectancy is that as the number of possible ply orientations is increased uniformly then the volume of the design space approaches an asymptotic limit. This limit is expected to be the same volume that would be obtained if all possible continuous ply angles were used. By doing such a study and by monitoring convergence, it can be shown that the approximate volume of the feasible region, denoted (Vol_j) where j = A, D, for in-plane and out-of-plane lamination parameters for an unrestricted design envelope respectively, is 3.2890 (confirmed by Setoodeh et al. (2006)). As such the constraining relationships between lamination parameters significantly reduce the potential design space from 16 to 3.289.

Table 2.1 (Feasible region volume for in-plane and out-of-plane lamination parameters)

Set of ply orientation (in degrees)	**Number of constraints**	**Volume of feasible region** (Vol_j^*)	**Relative volume (%)**
$[0, 90, \pm 45]$	4	1.3333	41
$[0, 90, \pm 45, \pm 30, \pm 60]$	20	1.5774	48
$[0, 90, \pm 45, \pm 15, \pm 30, \pm 60, \pm 75]$	54	2.5981	79
5 Degree Increments between [-85, 90]	594	3.2071	97.5
2.5 Degree Increments	2484	3.2690	99.4
1 Degree Increments	15930	3.2865	99.92

It is noted that the volume of the feasible region of the in-plane lamination parameters is identical to the volume of the feasible region of out-of-plane lamination parameters for any set of ply orientations where there are no restrictions on the nature of the lay-up, e.g. the lay-up does not need to be balanced or symmetric.

By monitoring convergence, it can be shown that the approximate volume of the feasible region, denoted (Vol_B), for the four coupling lamination parameters for any possible set of ply orientations is 6.6469.

Table 2.2 (Feasible region volume for coupling lamination parameters)

Set of ply orientation (in degrees)	Number of constraints	Volume of feasible region (Vol_B^*)	Relative volume (%)
$[0, 90, \pm 45]$	8	3.3333	50
$[0, 90, \pm 45, \pm 30, \pm 60]$	88	3.7704	57

$[0, 90, \pm 45, \pm 15, \pm 30, \pm 60, \pm 75]$	102	5.3207	80
5 Degree Increments between [-85, 90]	1170	6.4896	97.6
2.5 Degree Increments	4932	6.6085	99.4
1 Degree Increments	31770	6.6421	99.93

It can clearly be seen from Tables 2.1 and 2,2 that as the number of distinct ply orientations increases then so does the number of constraints on the feasible region of lamination parameters. Interestingly, the total number of constraints on the feasible region considering all lamination parameters is calculated as,

$$N_{ABD} = N_A + 3N_B + N_D + N_{ABD}^1 \qquad \textbf{(2.55)}$$

where N_{ABD} is the total number of constraints on the feasible region, N_{ABD}^1 is the number of constraints relating to one particular index on the *A*, *B* and *D* space. N_A is the number of constraints on the in-plane feasible region and others defined accordingly. The term $3N_B$ includes the constraints on the coupling feasible region as well as those relating the in-plane to out-of-plane to coupling lamination parameters. This result directly follows the results derived in this chapter. Additionally, from Tables 2.1 and 2.2 it is observed that the feasible region of lamination parameters can be approximated to within a 2.5% margin of the maximum obtainable volume, using a finite set of ply orientations. This feasible region consists of 5 degree increments between [-90, 90] for the ply orientations.

Interestingly, for an orthotropic symmetric laminate, ξ_3^A, ξ_4^A, ξ_3^D, $\xi_4^D = 0$ the area (since the feasible region is two dimensional) of the in-plane or out-of-plane feasible region (ξ_1^A, ξ_2^A and ξ_1^D, ξ_2^D respectively) is 2.666. The feasible region is well approximated by a set of $0, 90, \pm 30, \pm 45, \pm 60$ degree plies which has a corresponding area of 2.5. For this particular set of plies there is a 6.2 % relative error by using such few ply orientations, which suggests for design of orthotropic laminates then only a few ply orientations is required to obtain almost any combination of stiffness values. In summary, the information regarding the volume of the feasible regions may be used to determine the number of unique ply orientations to obtain specific types of anisotropic elastic response and so could be useful for elastic tailoring purposes. Once the set of ply orientations is determined, the method detailed in the chapter can readily be used to calculate all the constraints on feasible region of lamination parameters.

2.7 Conclusions

A method to derive the feasible region of lamination parameters for any predefined finite set of ply orientations has been presented. The work detailed generalises several approaches presented in the literature. The presented method was used to rederive and confirm the feasible region for $0, 90, \pm 45$ degree plies. Additionally, the feasible region for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies was derived, validated and confirmed.

The information detailed in this Chapter on the nature of the feasible region should prove useful for elastic tailoring purposes. For example, the volume of the feasible region can be studied and used to determine the set of ply orientations required to gain certain anisotropic elastic properties. Furthermore, a small finite set of ply orientations may enable to designer to achieve similar elastic properties compared to a continuous set of ply orientations.

Finally, as the method detailed herein can be used to determine the set of constraints on the feasible region of lamination parameters, such information can be used in conjunction with additional structural constraints and used in efficient optimization routines for the optimization of laminated composite structures. This is discussed and introduced in Chapter 3.

Chapter 3
Optimization Strategy

3.1 Introduction

In Chapter 2, a method to determine the feasible region of lamination parameters was presented. The method was used to analytically determine the boundary of the in-plane, coupling and out-of-plane feasible regions. The feasible region can be effectively utilized in efficient optimization routines in the design of laminated composite structures. In this Chapter, the optimization strategy is presented and discussed in detail.

A two-level optimization approach is used. At the first level, lamination parameters and plate thickness are used as the continuous design variables. A numerical gradient based method is then used to minimize the mass (and hence thickness) the laminated composite plate subject to buckling, strength (allowable laminate strain) and lamination parameter feasible region constraints. The first level determines the minimum thickness of the laminate for a given geometry and loading conditions. At the second level, ply orientations are used to determine a laminate stacking sequence which satisfies the buckling and strength (allowable laminate strain) constraints. As the thickness is determined at the first level, the objective of the second level is to use a discrete optimizer to determine a feasible lay-up which satisfies the set of design constraints.

3.2 Background

Lay-up optimization of laminated composites has evolved significantly over the past 25 years. Ghiassi (2009) recently provided a comprehensive review of the various optimization techniques which have been successfully applied to laminated composite design optimization. Recent focus has been on the optimization of laminated composites using lamination parameters and/or meta-heuristic approaches. In brief, meta-heuristics incorporate a heuristic idea to solve a type or class of computational optimization problem. In particular, these have been inspired by natural process such as evolution and colony based behaviour (Engelbrecht (2003)). A two-level optimization strategy employing lamination parameters, mathematical programming and GAs, was initially proposed by Yamazaki (1996). The optimization was split into two levels. Firstly, a gradient-based optimization was performed using the lamination parameters as design variables. At the second level, the square of the geometric distance between the target and current lamination parameters under consideration was minimized. The two level approach has recently been adopted by Herencia et al. (2007a-c, 2008a, 2008b) to solve the stacking sequence problem. At the first level, gradient based methods were used to determine optimal lamination parameters and plate thicknesses. All constraints such as strength and buckling were embedded at this level and necessary trade-offs considered. Lamination parameters are particularly useful intermediate design variables in the optimization of laminated composites because the constraining relationships between lamination parameters form a convex feasible region as discussed in Chapter 2. Consequently, where the objective function and constraints are a convex function (in minimization problems) of the design variables, gradient based methods guarantee that global optima are obtained (Bertsekas et al 2003). At the second level of the optimization a meta-heuristic optimizer is used to obtain a stacking sequence which satisfies the set of design constraints. It is important to note that at the second level the fitness function is highly non-convex. The non-convexity of the fitness function arises due to the mapping between ply orientations, lamination parameters and the fitness function. As such, gradient based methods may only find local optima and thus not entirely appropriate. The chief benefit of using a meta-heuristic algorithm is that gradient information is not required. Whilst local optima remain a problem, the ability to escape local optima can be studied and certain parameters adjusted to improve the performance of each method. Interestingly, Foldager et al. (1998) presented an alternative approach using ply orientations as design variables, where it was asserted ascertain that the mapping between ply orientations and the objective function maintained convexity. Despite this, the approach was deemed to be too inefficient to be viable. Furthermore, meta-heuristic methods can be adapted for discrete variable problems, unlike gradient based methods. Herencia et al. (2008a, 2008b) proposed two different solutions to the second level problem. Initially, the authors proposed a solution similar to Yamazaki (1996) where the square of geometric distance between the optimum and current lamination parameters was used,

$$F(\bar{\xi}) = \sum_{i=1}^{n} w_i (\xi_i^{opt} - \xi_i)^2 \tag{3.1}$$

where w_i are user defined weighting factors and n is the number of lamination parameters. It is observed that it was often difficult to match the optimum lamination parameters from the first level, especially for thinner laminates. By increasing the thickness, often by one or two plies, the authors were able to reduce the value of the fitness function. This is because the more plies, the greater then range of feasible lamination vectors assuming fixed ply thickness and a finite set of ply orientations. Recently, Herencia et al. (2008b) proposed an alternative to the second level objective function involving the design constraints. The design constraints are linearly approximated by a Taylor expansion (about the optimum point determined at the first level) for each design constraint. The objective (fitness) function, which is to be minimized, is then,

$$F(\bar{\xi}) = \max_i (G_i(\bar{\xi})) \tag{3.2}$$

where G_i is the set of second level constraints. As the constraints detailed in this Chapter are closed form solutions (CFS), G_i in Eqn. (3.2) is replaced with the CFS as the evaluation of these functions is efficient and accurate. Note the design constraints used in this thesis are formalized as inequality constraints. The two level strategy undertaken in this thesis is a natural extension of that proposed and used by Herencia et al. (2008a). The two level optimization strategy is outlined in Fig. 3.1.

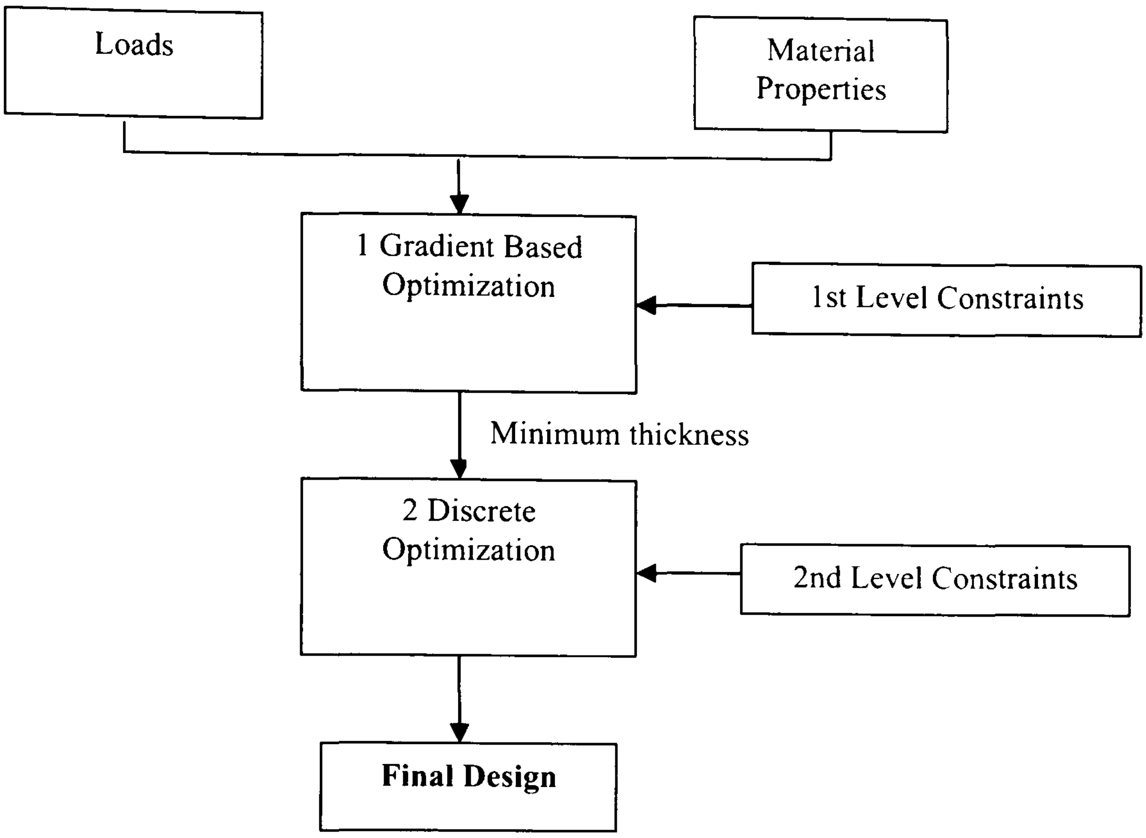

Fig. 3.1 – Two Level Optimization Strategy

In Fig. 3.1, the first step of the process is to feed the loads and material properties into the optimizer. Next, the 1st level optimization (and constraints) is initiated.

3.3 Continuous Optimization

In the continuous optimization, mathematical programming (MP) is used to minimize the objective function subject to a set of constraints. Mathematically, this is formulated as follows.

Minimize:

$$f(\bar{x}) \tag{3.3}$$

subject to:

$$G_j(\bar{x}) \leq 0 \tag{3.4}$$

where:

$$\bar{x}_k^l \leq \bar{x}_k \leq \bar{x}_k^u \tag{3.5}$$

Note, $f(\bar{x})$ is the objective function, $G_j(\bar{x})$ is the jth constraint and $\bar{x}_k^l$ and $\bar{x}_k^u$ are the lower and upper bounds on the kth design variable, respectively.

3.3.1 Objective Function

The objective function, which is to be minimized, is the mass of the structure.

$$f(\bar{x}) = \rho \sum_{i=1}^{n} a_i b_i t_i \tag{3.6}$$

where ρ is the density of the material, a_i is the length of the ith plate, b_i is the width ith plate and t_i is the ith thickness. Additionally, n is the number of plate elements in the structure. Where only one plate is under consideration (as is the case in this thesis), the objective function simplifies to,

$$f(\bar{x}) = \rho \cdot a \cdot A \tag{3.7}$$

3.3.2 Design Variables

In the optimization, the design variables are the local lamination parameters and thicknesses,

$$\bar{x}_i = \left(t,\ \xi_1^A,\ \xi_2^A,\ \xi_3^A,\ \xi_4^A,\ \xi_1^D,\ \xi_2^D,\ \xi_3^D,\ \xi_4^D \right)_i \tag{3.8}$$

where i references the ith plate. Again, this readily simplifies if only one plate is under consideration.

3.3.3 Design Constraints

The objective function is minimized subject to a set of inequality constraints. Although no equality constraints are used, the gradient optimizer replaces the inequality constraints with equality constraints and slack variables. Note the constraints are the feasible region of lamination parameters, strength constraints (allowable laminate strain) and buckling failure constraints.

In the optimization of laminated composite structures, several authors have integrated practical design constraints into the problem formulation. Such authors include Herencia et al. (2007a) and Liu and Haftka (2004). Common practical design constraints include a 10% rule and four ply rule for laminate stacking sequences. The 10% rule exists chiefly for $0, 90, \pm 45$ degree plies. This particular rule ensures a minimum strength in the principal fibre directions. However, for larger sets of ply orientations, the appropriateness of this rule is questioned. Furthermore, if the designer had ten different ply orientations and the 10% was adopted, the laminate strength properties would be fixed and the designer could change only the stacking sequence. Furthermore, this may lead to unnecessary weight penalties. Additionally, standard practice is to adopt a 4 ply rule to avoid large matrix cracking. However, the focus of this theiss is to evaluate the functional gains by utlizing an expanded set of ply orientations. As such, it is important to note that this thesis *does not* consider practical design constraints such as the 10% rule and four ply rule. Next, the design constraints used in this thesis are outlined.

3.3.3.1 Lamination Parameter Constraints

In Chapter 2 a method was given to determine the feasible region of lamination parameters for a given finite set of ply orientations. It was shown that four separate calculations were necessary.

1) Calculate the feasible region of in-plane lamination parameters
2) Calculate the feasible region of coupling lamination parameters
3) Calculate the feasible region of out-of-plane lamination parameters
4) Calculate the feasible region interrelating the in-plane, coupling and out-of-plane feasible regions.

In Chapter 2, it was stated that 60 degree plies were shown to be beneficial for shear buckling in long anisotropic plates (Weaver 2006). Additionally, 30 degree plies have been shown to be beneficial for aeroelastic purposes (Canale et al. 2009). Motivated by this, this thesis considers the expanded set of ply orientations $0, 90, \pm 30, \pm 45, \pm 60$. The following details the constraints on the in-plane, out-of-plane and coupling lamination parameters for this set of ply orientations calculated using the method outlined in Chapter 2.

Explicit expressions relating in-plane and relating out-of-plane lamination parameters:

$$\begin{pmatrix}
\frac{3}{2} & -\frac{1}{2} & \frac{1}{2} & -\frac{1}{2} & -1 \\
\frac{3}{2} & -\frac{1}{2} & -\frac{1}{2} & \frac{1}{2} & -1 \\
3-\sqrt{3} & \sqrt{3}-2 & \sqrt{3} & -1 & -1 \\
3-\sqrt{3} & \sqrt{3}-2 & -\sqrt{3} & 1 & -1 \\
1 & -1 & 0 & 0 & -1 \\
\frac{3-\sqrt{3}}{2} & \frac{\sqrt{3}-1}{2} & \frac{\sqrt{3}+1}{2} & \frac{1-\sqrt{3}}{2} & -1 \\
\frac{3-\sqrt{3}}{2} & \frac{\sqrt{3}-1}{2} & \frac{-\sqrt{3}-1}{2} & \frac{\sqrt{3}-1}{2} & -1 \\
0 & -1 & 0 & -\frac{1}{\sqrt{3}} & -1 \\
0 & -1 & 0 & \frac{1}{\sqrt{3}} & -1 \\
0 & 1 & 0 & -\sqrt{3} & -1 \\
0 & 1 & \sqrt{3} & 0 & -1 \\
0 & 1 & -\sqrt{3} & 0 & -1 \\
0 & 1 & 0 & \sqrt{3} & -1 \\
\frac{\sqrt{3}-3}{2} & \frac{\sqrt{3}-1}{2} & \frac{\sqrt{3}+1}{2} & \frac{\sqrt{3}-1}{2} & -1 \\
\frac{\sqrt{3}-3}{2} & \frac{\sqrt{3}-1}{2} & \frac{-\sqrt{3}-1}{2} & \frac{1-\sqrt{3}}{2} & -1 \\
-1 & -1 & 0 & 0 & -1 \\
\sqrt{3}-3 & \sqrt{3}-2 & 1-\sqrt{3} & -1 & -1 \\
\sqrt{3}-3 & \sqrt{3}-2 & \sqrt{3}-1 & 1 & -1 \\
-\frac{3}{2} & -\frac{1}{2} & \frac{1}{2} & \frac{1}{2} & -1 \\
-\frac{3}{2} & -\frac{1}{2} & -\frac{1}{2} & -\frac{1}{2} & -1
\end{pmatrix}
\begin{pmatrix} \xi_1^j \\ \xi_2^j \\ \xi_3^j \\ \xi_4^j \\ 1 \end{pmatrix} \leq 0 \qquad \textbf{(3.9)}$$

where j = A, D

Explicit expressions relating coupling lamination parameters

$$
\begin{pmatrix}
1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & -1 \\
1 & \frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & -1 \\
1 & 2-\sqrt{3} & 2\sqrt{3}-3 & \frac{1}{\sqrt{3}} & -1 \\
1 & 2-\sqrt{3} & 3-2\sqrt{3} & -\frac{1}{\sqrt{3}} & -1 \\
1 & \frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & -1 \\
1 & \frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & -1 \\
1 & -\frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & -1 \\
1 & -\frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & -1 \\
1 & \sqrt{3}-2 & 2\sqrt{3}-3 & -\frac{1}{\sqrt{3}} & -1 \\
1 & \sqrt{3}-2 & 3-2\sqrt{3} & \frac{1}{\sqrt{3}} & -1 \\
1 & -\frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & -1 \\
1 & -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & -1 \\
\frac{6-2\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & \frac{2}{3} & -1 \\
\frac{6-2\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & -1 \\
\frac{6-2\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & \frac{2}{3} & -1 \\
\frac{6-2\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & -1 \\
\sqrt{3}-1 & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & \sqrt{3}-1 & -1 \\
\sqrt{3}-1 & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & 1-\sqrt{3} & -1 \\
\sqrt{3}-1 & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & 1-\sqrt{3} & -1 \\
\sqrt{3}-1 & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & \sqrt{3}-2 & -1
\end{pmatrix}
\begin{pmatrix} \xi_1^B \\ \xi_2^B \\ \xi_3^B \\ \xi_4^B \\ 1 \end{pmatrix} \le 0 \tag{3.10}
$$

$$
\begin{pmatrix}
\frac{2}{3} & \frac{2}{3} & 0 & 0 & -1 \\
\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -1 \\
\frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -1 \\
\frac{3-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -1 \\
\frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -1 \\
\frac{3-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -1 \\
\frac{4\sqrt{3}-6}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4\sqrt{3}-4}{3} & \frac{2\sqrt{3}-4}{3} & -1 \\
\frac{4\sqrt{3}-6}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4-4\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & -1 \\
\frac{4\sqrt{3}-6}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4\sqrt{3}-4}{3} & \frac{4-2\sqrt{3}}{3} & -1 \\
\frac{4\sqrt{3}-6}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4-4\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & -1 \\
2-\sqrt{3} & 2-\sqrt{3} & 1 & \frac{3-2\sqrt{3}}{3} & -1 \\
2-\sqrt{3} & 2-\sqrt{3} & -1 & \frac{2\sqrt{3}-3}{3} & -1 \\
2-\sqrt{3} & \sqrt{3}-2 & 1 & \frac{3-2\sqrt{3}}{3} & -1 \\
2-\sqrt{3} & \sqrt{3}-2 & -1 & \frac{2\sqrt{3}-3}{3} & -1 \\
0 & 1 & 0 & \frac{1}{\sqrt{3}} & -1 \\
0 & 1 & 0 & -\frac{1}{\sqrt{3}} & -1 \\
0 & \frac{2}{3} & \frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -1 \\
0 & \frac{2}{3} & \frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -1 \\
0 & \frac{2}{3} & 0 & \frac{2}{\sqrt{3}} & -1 \\
0 & \frac{2}{3} & 0 & -\frac{2}{\sqrt{3}} & -1
\end{pmatrix}
\begin{pmatrix}
\xi_1^B \\ \xi_2^B \\ \xi_3^B \\ \xi_4^B \\ 1
\end{pmatrix} \le 0 \qquad \textbf{(3.11)}
$$

$$
\begin{pmatrix}
0 & \frac{2}{3} & -\frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -1 \\
0 & \frac{2}{3} & -\frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -1 \\
0 & 4-2\sqrt{3} & 4\sqrt{3}-6 & 0 & -1 \\
0 & 4-2\sqrt{3} & 6-4\sqrt{3} & 0 & -1 \\
0 & 2\sqrt{3}-4 & 4\sqrt{3}-6 & 0 & -1 \\
0 & 2\sqrt{3}-4 & 6-4\sqrt{3} & 0 & -1 \\
0 & -\frac{2}{3} & \frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -1 \\
0 & -\frac{2}{3} & \frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -1 \\
0 & -\frac{2}{3} & 0 & \frac{2}{\sqrt{3}} & -1 \\
0 & -\frac{2}{3} & 0 & -\frac{2}{\sqrt{3}} & -1 \\
0 & -\frac{2}{3} & -\frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -1 \\
0 & -\frac{2}{3} & -\frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -1 \\
0 & -1 & 0 & \frac{1}{\sqrt{3}} & -1 \\
0 & -1 & 0 & -\frac{1}{\sqrt{3}} & -1 \\
\sqrt{3}-2 & 2-\sqrt{3} & 1 & \frac{2}{\sqrt{3}}-1 & -1 \\
\sqrt{3}-2 & 2-\sqrt{3} & -1 & 1-\frac{2}{\sqrt{3}} & -1 \\
\sqrt{3}-2 & \sqrt{3}-2 & 1 & 1-\frac{2}{\sqrt{3}} & -1 \\
\sqrt{3}-2 & \sqrt{3}-2 & -1 & \frac{2}{\sqrt{3}}-1 & -1 \\
\frac{6-4\sqrt{3}}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4\sqrt{3}-4}{3} & \frac{4-2\sqrt{3}}{3} & -1 \\
\frac{6-4\sqrt{3}}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4-4\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & -1
\end{pmatrix}
\begin{pmatrix} \xi_1^B \\ \xi_2^B \\ \xi_3^B \\ \xi_4^B \\ 1 \end{pmatrix} \leq 0 \qquad \textbf{(3.12)}
$$

$$\begin{pmatrix}
\frac{6-4\sqrt{3}}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4\sqrt{3}-4}{3} & \frac{2\sqrt{3}-4}{3} & -1 \\
\frac{6-4\sqrt{3}}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4-4\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & -1 \\
\frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -1 \\
\frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-1}{3} & \frac{-1-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -1 \\
\frac{\sqrt{3}-3}{3} & \frac{1-\sqrt{3}}{3} & \frac{1+\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -1 \\
\frac{\sqrt{3}-3}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -1 \\
-\frac{2}{3} & \frac{2}{3} & 0 & 0 & -1 \\
-\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -1 \\
1-\sqrt{3} & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & 1-\sqrt{3} & -1 \\
1-\sqrt{3} & \frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-3}{3} & \sqrt{3}-1 & -1 \\
1-\sqrt{3} & \frac{\sqrt{3}-3}{3} & \frac{3-\sqrt{3}}{3} & \sqrt{3}-1 & -1 \\
1-\sqrt{3} & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & 1-\sqrt{3} & -1 \\
\frac{2\sqrt{3}-6}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & -1 \\
\frac{2\sqrt{3}-6}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2-2\sqrt{3}}{3} & \frac{2}{3} & -1 \\
\frac{2\sqrt{3}-6}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & -1 \\
\frac{2\sqrt{3}-6}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & \frac{2}{3} & -1 \\
-1 & \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & -1 \\
-1 & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & -1 \\
-1 & 2-\sqrt{3} & 2\sqrt{3}-3 & -\frac{1}{\sqrt{3}} & -1 \\
-1 & 2-\sqrt{3} & 3-2\sqrt{3} & \frac{1}{\sqrt{3}} & -1
\end{pmatrix}
\begin{pmatrix} \xi_1^B \\ \xi_2^B \\ \xi_3^B \\ \xi_4^B \\ 1 \end{pmatrix} \le 0 \qquad \textbf{(3.13)}$$

$$
\begin{pmatrix}
-1 & \frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & -1 \\
-1 & \frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & -1 \\
-1 & -\frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & -1 \\
-1 & -\frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & -1 \\
-1 & \sqrt{3}-2 & 2\sqrt{3}-3 & \frac{1}{\sqrt{3}} & -1 \\
-1 & \sqrt{3}-2 & 3-2\sqrt{3} & -\frac{1}{\sqrt{3}} & -1 \\
-1 & \frac{-1}{3} & \frac{1}{3} & \frac{1}{3} & -1 \\
-1 & \frac{-1}{3} & \frac{-1}{3} & \frac{-1}{3} & -1
\end{pmatrix}
\begin{pmatrix} \xi_1^B \\ \xi_2^B \\ \xi_3^B \\ \xi_4^B \\ 1 \end{pmatrix} \le 0 \qquad \textbf{(3.14)}
$$

Note, the boundary of the feasible region of lamination parameters for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies is readily obtained from Algorithms 2.1 and 2.2 detailed in Chapter 2. Each row of the matrices (M_1-M_5), shown below, corresponds to a unique combination of h_1, h_2, h_3, h_4, H^L, H^U as discussed in detail in Chapter 2.

$$
M_1 = \begin{pmatrix}
1 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & -\frac{2}{3} & \frac{4}{3} \\
1 & \frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & -\frac{2}{3} & \frac{4}{3} \\
1 & 2-\sqrt{3} & 2\sqrt{3}-3 & \frac{1}{\sqrt{3}} & 1-\sqrt{3} & 3-\sqrt{3} \\
1 & 2-\sqrt{3} & 3-2\sqrt{3} & -\frac{1}{\sqrt{3}} & 1-\sqrt{3} & 3-\sqrt{3} \\
1 & \frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & \frac{-1}{3-\sqrt{3}} & \frac{4+\sqrt{3}}{3+\sqrt{3}} \\
1 & \frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & \frac{-1}{3-\sqrt{3}} & \frac{4+\sqrt{3}}{3+\sqrt{3}} \\
1 & -\frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & \frac{-4-\sqrt{3}}{3+\sqrt{3}} & \frac{1}{3-\sqrt{3}} \\
1 & -\frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & \frac{-4-\sqrt{3}}{3+\sqrt{3}} & \frac{1}{3-\sqrt{3}} \\
1 & \sqrt{3}-2 & 2\sqrt{3}-3 & -\frac{1}{\sqrt{3}} & \sqrt{3}-3 & \sqrt{3}-1 \\
1 & \sqrt{3}-2 & 3-2\sqrt{3} & \frac{1}{\sqrt{3}} & \sqrt{3}-3 & \sqrt{3}-1 \\
1 & -\frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & -\frac{4}{3} & \frac{2}{3} \\
1 & -\frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{6-2\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & \frac{2}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{6-2\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{6-2\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & \frac{2}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{6-2\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & -\frac{4}{3} & \frac{2}{3} \\
\sqrt{3}-1 & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & \sqrt{3}-1 & \frac{2\sqrt{3}-6}{3} & \frac{2}{\sqrt{3}} \\
\sqrt{3}-1 & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & 1-\sqrt{3} & \frac{2\sqrt{3}-6}{3} & \frac{2}{\sqrt{3}} \\
\sqrt{3}-1 & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & 1-\sqrt{3} & -\frac{2}{\sqrt{3}} & \frac{6-2\sqrt{3}}{3} \\
\sqrt{3}-1 & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & \sqrt{3}-2 & -\frac{2}{\sqrt{3}} & \frac{6-2\sqrt{3}}{3}
\end{pmatrix}
$$

$$M_2 = \begin{pmatrix}
\frac{2}{3} & \frac{2}{3} & 0 & 0 & -\frac{2}{3} & \frac{4}{3} \\
\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -\frac{4}{3} & \frac{2}{3} \\
\frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{3-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{3-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{4\sqrt{3}-6}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4\sqrt{3}-4}{3} & \frac{2\sqrt{3}-4}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{4\sqrt{3}-6}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4-4\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{4\sqrt{3}-6}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4\sqrt{3}-4}{3} & \frac{4-2\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{4\sqrt{3}-6}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4-4\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & -\frac{2}{3} & \frac{4}{3} \\
2-\sqrt{3} & 2-\sqrt{3} & 1 & \frac{3-2\sqrt{3}}{3} & \sqrt{3}-3 & \sqrt{3}-1 \\
2-\sqrt{3} & 2-\sqrt{3} & -1 & \frac{2\sqrt{3}-3}{3} & \sqrt{3}-3 & \sqrt{3}-1 \\
2-\sqrt{3} & \sqrt{3}-2 & 1 & \frac{3-2\sqrt{3}}{3} & 1-\sqrt{3} & 3-\sqrt{3} \\
2-\sqrt{3} & \sqrt{3}-2 & -1 & \frac{2\sqrt{3}-3}{3} & 1-\sqrt{3} & 3-\sqrt{3} \\
0 & 1 & 0 & \frac{1}{\sqrt{3}} & -1 & 1 \\
0 & 1 & 0 & -\frac{1}{\sqrt{3}} & -1 & 1 \\
0 & \frac{2}{3} & \frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{4}{3} & \frac{2}{3} \\
0 & \frac{2}{3} & \frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3} \\
0 & \frac{2}{3} & 0 & \frac{2}{\sqrt{3}} & -\frac{4}{3} & \frac{2}{3} \\
0 & \frac{2}{3} & 0 & -\frac{2}{\sqrt{3}} & -\frac{4}{3} & \frac{2}{3}
\end{pmatrix}$$

$$M_3=\begin{pmatrix}
0 & \frac{2}{3} & -\frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{4}{3} & \frac{2}{3}\\
0 & \frac{2}{3} & -\frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3}\\
0 & 4-2\sqrt{3} & 4\sqrt{3}-6 & 0 & 2-\sqrt{3} & 4-2\sqrt{3}\\
0 & 4-2\sqrt{3} & 6-4\sqrt{3} & 0 & 2-\sqrt{3} & 4-2\sqrt{3}\\
0 & 2\sqrt{3}-4 & 4\sqrt{3}-6 & 0 & 2\sqrt{3}-4 & \sqrt{3}-2\\
0 & 2\sqrt{3}-4 & 6-4\sqrt{3} & 0 & 2\sqrt{3}-4 & \sqrt{3}-2\\
0 & -\frac{2}{3} & \frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & \frac{4}{3}\\
0 & -\frac{2}{3} & \frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3}\\
0 & -\frac{2}{3} & 0 & \frac{2}{\sqrt{3}} & -\frac{2}{3} & \frac{4}{3}\\
0 & -\frac{2}{3} & 0 & -\frac{2}{\sqrt{3}} & -\frac{2}{3} & \frac{4}{3}\\
0 & -\frac{2}{3} & -\frac{2}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & \frac{4}{3}\\
0 & -\frac{2}{3} & -\frac{2}{3} & \frac{2-2\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3}\\
0 & -1 & 0 & \frac{1}{\sqrt{3}} & -1 & 1\\
0 & -1 & 0 & -\frac{1}{\sqrt{3}} & -1 & 1\\
\sqrt{3}-2 & 2-\sqrt{3} & 1 & \frac{2}{\sqrt{3}}-1 & \sqrt{3}-3 & \sqrt{3}-1\\
\sqrt{3}-2 & 2-\sqrt{3} & -1 & 1-\frac{2}{\sqrt{3}} & \sqrt{3}-3 & \sqrt{3}-1\\
\sqrt{3}-2 & \sqrt{3}-2 & 1 & 1-\frac{2}{\sqrt{3}} & 1-\sqrt{3} & 3-\sqrt{3}\\
\sqrt{3}-2 & \sqrt{3}-2 & -1 & \frac{2}{\sqrt{3}}-1 & 1-\sqrt{3} & 3-\sqrt{3}\\
\frac{6-4\sqrt{3}}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4\sqrt{3}-4}{3} & \frac{4-2\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3}\\
\frac{6-4\sqrt{3}}{3} & \frac{8-4\sqrt{3}}{3} & \frac{4-4\sqrt{3}}{3} & \frac{2\sqrt{3}-4}{3} & -\frac{4}{3} & \frac{2}{3}
\end{pmatrix}$$

$$
M_4 = \begin{pmatrix}
\frac{6-4\sqrt{3}}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4\sqrt{3}-4}{3} & \frac{2\sqrt{3}-4}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{6-4\sqrt{3}}{3} & \frac{4\sqrt{3}-8}{3} & \frac{4-4\sqrt{3}}{3} & \frac{4-2\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-1}{3} & \frac{1+\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-1}{3} & \frac{-1-\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{\sqrt{3}-3}{3} & \frac{1-\sqrt{3}}{3} & \frac{1+\sqrt{3}}{3} & \frac{1-\sqrt{3}}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{\sqrt{3}-3}{3} & \frac{1-\sqrt{3}}{3} & \frac{-1-\sqrt{3}}{3} & \frac{\sqrt{3}-1}{3} & -\frac{2}{3} & \frac{4}{3} \\
-\frac{2}{3} & \frac{2}{3} & 0 & 0 & -\frac{2}{3} & \frac{4}{3} \\
-\frac{2}{3} & -\frac{2}{3} & 0 & 0 & -\frac{4}{3} & \frac{2}{3} \\
1-\sqrt{3} & \frac{3-\sqrt{3}}{3} & \frac{3-\sqrt{3}}{3} & 1-\sqrt{3} & \frac{2\sqrt{3}-6}{3} & \frac{2}{\sqrt{3}} \\
1-\sqrt{3} & \frac{3-\sqrt{3}}{3} & \frac{\sqrt{3}-3}{3} & \sqrt{3}-1 & \frac{2\sqrt{3}-6}{3} & \frac{2}{\sqrt{3}} \\
1-\sqrt{3} & \frac{\sqrt{3}-3}{3} & \frac{3-\sqrt{3}}{3} & \sqrt{3}-1 & -\frac{2}{\sqrt{3}} & \frac{6-2\sqrt{3}}{3} \\
1-\sqrt{3} & \frac{\sqrt{3}-3}{3} & \frac{\sqrt{3}-3}{3} & 1-\sqrt{3} & -\frac{2}{\sqrt{3}} & \frac{6-2\sqrt{3}}{3} \\
\frac{2\sqrt{3}-6}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{2\sqrt{3}-6}{3} & \frac{4-2\sqrt{3}}{3} & \frac{2-2\sqrt{3}}{3} & \frac{2}{3} & -\frac{2}{3} & \frac{4}{3} \\
\frac{2\sqrt{3}-6}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2\sqrt{3}-2}{3} & -\frac{2}{3} & -\frac{4}{3} & \frac{2}{3} \\
\frac{2\sqrt{3}-6}{3} & \frac{2\sqrt{3}-4}{3} & \frac{2-2\sqrt{3}}{3} & \frac{2}{3} & -\frac{4}{3} & \frac{2}{3} \\
-1 & \frac{1}{3} & \frac{1}{3} & -\frac{1}{3} & -\frac{2}{3} & \frac{4}{3} \\
-1 & \frac{1}{3} & -\frac{1}{3} & \frac{1}{3} & -\frac{2}{3} & \frac{4}{3} \\
-1 & 2-\sqrt{3} & 2\sqrt{3}-3 & -\frac{1}{\sqrt{3}} & 1-\sqrt{3} & 3-\sqrt{3} \\
-1 & 2-\sqrt{3} & 3-2\sqrt{3} & \frac{1}{\sqrt{3}} & 1-\sqrt{3} & 3-\sqrt{3}
\end{pmatrix}
$$

$$
M_5 = \begin{pmatrix}
-1 & \frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & -\frac{1}{3-\sqrt{3}} & \frac{4+\sqrt{3}}{3+\sqrt{3}} \\
-1 & \frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & -\frac{1}{3-\sqrt{3}} & \frac{4+\sqrt{3}}{3+\sqrt{3}} \\
-1 & -\frac{1}{3+\sqrt{3}} & \frac{1}{\sqrt{3}} & \frac{1-\sqrt{3}}{2} & \frac{-4-\sqrt{3}}{3+\sqrt{3}} & \frac{1}{3-\sqrt{3}} \\
-1 & -\frac{1}{3+\sqrt{3}} & -\frac{1}{\sqrt{3}} & \frac{\sqrt{3}-1}{2} & \frac{-4-\sqrt{3}}{3+\sqrt{3}} & \frac{1}{3-\sqrt{3}} \\
-1 & \sqrt{3}-2 & 2\sqrt{3}-3 & \frac{1}{\sqrt{3}} & \sqrt{3}-3 & 1-\sqrt{3} \\
-1 & \sqrt{3}-2 & 3-2\sqrt{3} & -\frac{1}{\sqrt{3}} & \sqrt{3}-3 & 1-\sqrt{3} \\
-1 & -\frac{1}{3} & \frac{1}{3} & \frac{1}{3} & -\frac{4}{3} & \frac{2}{3} \\
-1 & -\frac{1}{3} & -\frac{1}{3} & -\frac{1}{3} & -\frac{4}{3} & \frac{2}{3} \\
1 & 0 & 0 & 0 & -1 & 1 \\
0 & 1 & 0 & 0 & -1 & 1 \\
0 & 0 & 1 & 0 & -1 & 1
\end{pmatrix}
$$

Next, let,

$$
A = \begin{pmatrix} M_1 \\ M_2 \\ M_3 \\ M_4 \\ M_5 \end{pmatrix} \tag{3.15}
$$

and,

$$
\begin{aligned}
X_L^A &= \begin{pmatrix} \xi_1^A & \xi_2^A & \xi_3^A & \xi_4^A & -1 & 0 \end{pmatrix}^T \\
X_U^A &= \begin{pmatrix} \xi_1^A & \xi_2^A & \xi_3^A & \xi_4^A & 0 & -1 \end{pmatrix}^T \\
X_L^D &= \begin{pmatrix} \xi_1^D & \xi_2^D & \xi_3^D & \xi_4^D & -1 & 0 \end{pmatrix}^T \\
X_U^D &= \begin{pmatrix} \xi_1^D & \xi_2^D & \xi_3^D & \xi_4^D & 0 & -1 \end{pmatrix}^T \\
X^B &= \begin{pmatrix} \xi_1^B & \xi_2^B & \xi_3^B & \xi_4^B & 0 & 0 \end{pmatrix}^T
\end{aligned} \tag{3.16}
$$

Then it follows from Eqns. (3.16) (with $k = 1$) that,

$$\left(AX_L^A\right)_i^4 + 3\left(AX^B\right)_i^2 - 4\left(AX_L^A\right)_i\left(AX_L^D\right)_i \le 0$$
$$\left(AX_U^A\right)_i^4 + 3\left(AX^B\right)_i^2 - 4\left(AX_U^A\right)_i\left(AX_U^D\right)_i \le 0 \qquad \textbf{(3.17)}$$

where the subscript i is the element in the i^{th} row of the indicated matrix multiplication. Furthermore, $1 \le i \le 91$, indicating the number of rows in the matrix A. Additionally,

$$\left(\xi_4^A + \frac{\sqrt{3}}{2}\right)^4 + 3k^2(\xi_4^B)^2 - 4k^2\left(\xi_4^A + \frac{\sqrt{3}}{2}\right)\left(\xi_4^D + \frac{\sqrt{3}}{2}\right) \le 0$$
$$\left(\xi_4^A - \frac{\sqrt{3}}{2}\right)^4 + 3k^2(\xi_4^B)^2 - 4k^2\left(\xi_4^A - \frac{\sqrt{3}}{2}\right)\left(\xi_4^D - \frac{\sqrt{3}}{2}\right) \le 0 \qquad \textbf{(3.18)}$$

where $k = \dfrac{\sqrt{3}}{2}$. Eqns. (3.17) and (3.18) fully define the constraints on the interconnected feasible region of lamination parameters for angles restricted to $0, 90, \pm 30, \pm 45, \pm 60$ degree plies. In sum, Eqns. (3.9-3.14) in addition to Eqns. (3.17-3.18) fully define the feasible regions of lamination parameters for $0, 90, \pm 30, \pm 45, \pm 60$ degree plies.

Using the method outlined in Chapter 2, the set of constraints can be readily derived for any finite set of ply orientations. In Chapter 7 (numerical examples), the method outlined in Chapter 2 will be used to generate the set of constraints for a number of different sets of ply orientations. The effect of the size of the set of ply orientations will be investigated. However, due to space limitations these constraints will not be explicitly stated.

3.3.3.2 Failure Strength Constraints

Strains at laminate level are limited by an allowable strain value. Classical Laminate Theory (Tsai and Hahn 1980) is used to calculate the laminate strains for the composite plate, where,

$$\begin{bmatrix} \varepsilon_x^0 \\ \varepsilon_y^0 \\ \varepsilon_{xy}^0 \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{16} \\ A_{12} & A_{22} & A_{26} \\ A_{16} & A_{26} & A_{26} \end{bmatrix}^{-1} \begin{bmatrix} N_x \\ N_y \\ N_{xy} \end{bmatrix} \quad \textbf{(3.19)}$$

The strength reserve factor is given by the ratio between the allowable and the applied strain (calculated from Eqn. (3.19)). Hence,

$$RF_i^s = \frac{\varepsilon_{ai}}{\varepsilon_i^0} \quad \textbf{(3.20)}$$

where $i = x, y, xy$ and where s denotes strength. The strength constraint applied to the plate is given by,

$$\frac{1}{RF_i^s} - 1 \leq 0 \quad \textbf{(3.21)}$$

where $i = x, y, xy$. Note, The allowable strains are (in compression) -3600 (in x), -3600 (in y), shear -7200 (in xy) microstrain. Next the buckling constraint is discussed.

3.3.3.3 Buckling Constraint

Local buckling analysis is undertaken using closed form solutions (Weaver 2006). The plate is assumed to be flat, simply supported on all four edges and under compression and shear loads. Non-dimensionalized parameters (Weaver and Nemeth 2007) are used to calculate buckling coefficients in order to determine the critical buckling loads. The non-dimensionalized parameters are defined in terms of the out-of-plane stiffnesses,

$$\alpha = \sqrt[4]{\frac{D_{22}}{D_{11}}},\ \beta = \frac{D_{12} + 2D_{66}}{\sqrt{D_{11}D_{22}}},\ \gamma = \frac{D_{16}}{\sqrt[4]{D_{11}^3 D_{22}}},\ \delta = \frac{D_{26}}{\sqrt[4]{D_{11}D_{22}^3}} \quad \textbf{(3.22)}$$

The critical buckling load of an anisotropic plate which is simply supported on all four edges and under normal loading was approximated using the following expression (Weaver and Nemeth 2007),

$$N_x^{cr} = K_x \frac{\pi^2}{b^2} \sqrt{D_{11} D_{12}} \tag{3.23}$$

where K_x is a non-dimensional buckling coefficient given by,

$$K_x = 2(1+\beta) - 2(3+\beta+2\gamma^2)\frac{(\gamma+3\delta)^2}{(3+\beta)^2} - 4(\delta+2\gamma^3-\beta\gamma)\frac{(\gamma+3\delta)^3}{(3+\beta)^3} \tag{3.24}$$

It is noted that sufficient accuracy for K_x is obtained when $|\gamma|, |\delta| < 0.4$. If $|\gamma|, |\delta| > 0.4$ an iterative procedure is used to determine the value of K_x. For further details see Weaver and Nemeth 2007. The reserve factor for the uniaxial compression loading is given by,

$$RF_{Bx} = \frac{N_x^{cr}}{N_x} \tag{3.25}$$

where B denotes buckling. The shear buckling coefficient was defined in terms of the non-dimensional parameters (Weaver 2006),

$$K_{xy} = 3.42 + 2.05\beta - 0.13\beta^2 - 1.79\gamma - 6.89\delta + 0.36\beta(2\gamma+\delta) - 0.25(2\gamma+\delta)^2 \tag{3.26}$$

The critical shear buckling load is calculated as,

$$N_{xy}^{cr} = \frac{\pi^2}{b^2} K_{xy} \sqrt[4]{D_{11} D_{22}^3} \tag{3.27}$$

The reserve factor for the shear loading is thus,

$$RF_{Bxy} = \frac{N_{xy}^{cr}}{|N_{xy}|} \tag{3.28}$$

Note, in the case of negative shear, the shear buckling coefficient is calculated assuming that the sign of each ply angle is reversed. Lastly, the following formula (Weaver 2009a) is used to address the interaction between normal and shear buckling,

$$\frac{1}{RF_B} = \frac{1}{RF_{Bx}} + \frac{1}{(RF_{Bxy})^{(1.9+0.1\beta)}} \tag{3.29}$$

The buckling constraint is therefore,

$$1 - RF_B \leq 0 \tag{3.30}$$

Next, gradient based optimization steps are presented.

3.3.4 Mathematical Programming

In the previous section, it was stated that MP, and in particular, a gradient based optimization, is used to minimize the objective function subject to a set of constraints. To achieve this, the Lagrangian of the two components is formed. A Lagrangian approach is adopted to handle the objective function, *f,* and the constraints (linear and non-linear, equality G^{eq} and inequality G^{in}). The Lagrangian, *L* is formulated as,

minimize
$$L = f + \sum_{i=1}^{m} \lambda_i G_i^{eq} + \sum_{j=1}^{n} \mu_j G_j^{in} \tag{3.31}$$

such that,
$$\nabla L = \nabla f + \sum_{i=1}^{m} \lambda_i \nabla G_i^{eq} + \sum_{j=1}^{n} \mu_j \nabla G_j^{in} = 0 \tag{3.32}$$

and where (Primal feasibility condition),

$$\begin{aligned} G_i^{eq}(x) &= 0 \\ G_j^{in}(x) &<= 0 \end{aligned} \tag{3.33}$$

and (Dual feasibility condition),

$$\mu_j \geq 0 \tag{3.34}$$

Furthermore, the non-degeneracy condition is satisfied, if for all *i*

$$\nabla G_i \neq 0 \qquad \textit{where} \qquad G_i = 0 \tag{3.35}$$

Additionally, the complementary slackness property states,

$$\mu_j G_j^{in}(x) = 0 \;\forall\; j \qquad \textbf{(3.36)}$$

Eqns. (3.31-3.36) form the Karush-Kuhn-Tucker conditions (Bertsekas et al. 2003). As mentioned a gradient based approach is adopted. In particular a quasi-Newton approach is used. Note whilst a general Lagrangian formulation is given here, only inequality constraints are used in this thesis. The Lagrangian function is approximated as,

$$L(x_k + \Delta x) \approx L(x_k) + \nabla L(x_k)^T \Delta x + \frac{1}{2}\Delta x^T H \Delta x \qquad \textbf{(3.37)}$$

where H is an approximation to the Hessian (the matrix of second order derivatives). This is achieved using the BFGS (Bertsekas et al. 2003) method. This gradient based optimization is undertaken using the *fmincon* function in Optimization Toolbox of MATLAB (MATLAB 2009a). For further details on the gradient based approach, please see Appendix A. Next, the gradient calculations are discussed. The sensitivities with respect to the objective function are calculated as follows,

$$\frac{\partial f_i}{\partial x_j} = a_i b_i t_i \rho \qquad \textbf{(3.38)}$$

Sensitivities with respect to the structural constraints are,

$$\frac{\partial G_i}{\partial x_j} = \frac{\partial G_i}{\partial x_j} + \frac{\partial G_i}{\partial N_k}\frac{\partial N_k}{\partial x_j} \qquad \textbf{(3.39)}$$

Note, $\frac{\partial G_i}{\partial x_j}$ and $\frac{\partial G_i}{\partial N_k}$ are calculated using a forward finite difference approximation. Additionally, $\frac{\partial N_k}{\partial x_j}$, is computed using Finite Element Analysis (FEA) where the number of plates exceeds one, i.e. $i > 1$. Note for a single plate under applied continuous external loads, $\frac{\partial N_k}{\partial x_j}$ is zero and hence Eqn. (3.39) simplifies to $\frac{\partial G_i}{\partial x_j}$.

At the end of the first level, a minimum thickness is obtained with all constraints less than or equal to zero. The obtained continuous thickness is rounded up to the nearest ply (discrete). Note, by increasing the thickness, the reserve factors also increase marginally. At the end of the first level, rounded thicknesses, set of ply orientations and structural geometry are passed to the second level. If the second level constraints are not known in closed form, then optimal lamination parameters are passed to the second level to allow the construction of linear approximations. Once this stage is completed, the second level, which is the discrete optimizer, is launched.

3.4 Discrete Optimization

At the second level, ply orientations are used a design variables to determine a stacking sequence (of determined minimum thickness) which satisfies the set of design constraints. Formally this is a constraint satisfaction problem (CSP). At the second level, ply thickness is assumed to be fixed and the design variables are ply orientations selected from a given finite set of ply orientations. Various optimization techniques can be utilized at the second level and four commonly used approaches will be presented and analysed in Chapter 4. The second level optimization block structure is shown in Fig. 3.2

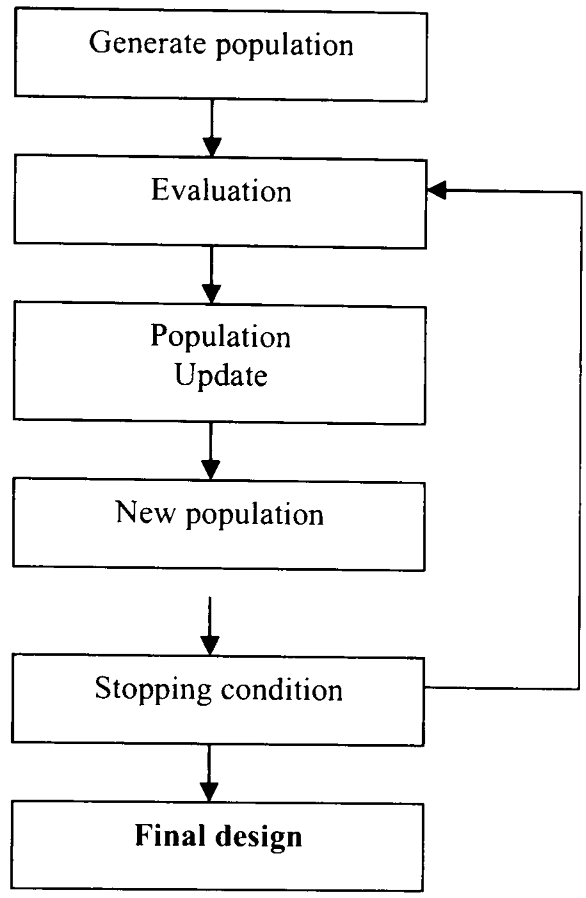

Fig. 3.2 - Discrete Optimizer Structure

3.4.1 Second Level Objective Function

The objective function, which is to be minimized is defined as,

$$F(\bar{\theta}) = \max_{i}(G_i(\bar{\theta})) \qquad \textbf{(3.40)}$$

where $G_i(\bar{\theta})$ is the ith constraint (which is a CFS). As the second level objective will be used in the context of heuristic optimizers, it will commonly be referred to as the fitness function. Note, whilst Herencia et al. (2008b) used linear approximations of the constraints (this where first order sensitivity information was obtained from the gradient optimizer to form a linear approximation) this thesis considers the full constraints as they are known in closed form.

3.4.2 Second Level Design Variables

The design variables are the set of predefined ply orientations. This is represented by a vector of angles, $\bar{\theta}$. Note, at the second level of the optimization a set greater than that used to derive the feasible region constraints (at the first level) is permissible. For example, if the feasible region was calculated using $0, 90, \pm 45$ degrees, the second level can include this set plus additional angles. This process leads to no loss in generality, however the inclusion of additional ply orientations may affect the efficiency of the selected algorithm, On the other hand, using a set of angles less than the set used to derive the feasible region may result in a solution (final design) not being obtained.

3.5 Conclusions

In this Chapter, the optimization strategy was presented and discussed in detail. A two-level optimization approach was introduced. At the first level, a gradient based method is used to minimize the Lagrangian (objective function in combination with the linear and non-linear constraints). Lamination parameters and plate thicknesses are used as design variables. Once the minimum mass of the plate is obtained, the second level is initialized. At the second level, a discrete optimizer is used to determine a stacking sequence which satisfies the set of design constraints. To achieve this, several methods can be adopted. In Chapter 4, a branch and bound (BB), genetic algorithm (GA), particle swarm optimization (PSO) and ant colony optimization (ACO) approaches are considered for this purpose. It will be shown that a GA, PSO and ACO offer a suitable path to stacking sequence optimization due to their heuristic nature. Following a formal analysis and numerical examples, it will be argued that PSO and ACO offer the best route to identifying laminate stacking sequences which solve the CSP.

Chapter 4

Analysis and Benchmarking of Optimization Methods to Determine Lay-Ups

4.1 Introduction

In Chapter 2, a method to determine the feasible region of lamination parameters was presented. In Chapter 3, a two-level optimization approach was introduced. At the first level, lamination parameters and plate thicknesses are used to minimize the mass of the composite structure subject to a set of design constraints including lamination parameter/feasible region constraints. At the second level, a discrete optimizer is used to identify a stacking sequence (where optimum thickness is obtained at the first level) which satisfies the set of design constraints. Note the second level does not seek to maximize a particular function, but rather searches to find a feasible solution. Chapters 2 and 3 embody the details required to the first level of the optimization. The goal of this Chapter is the analysis and benchmarking of the discrete optimizers used to identify stacking sequences at the second level of the optimization process.

Over the past 25 years, numerous optimization techniques have been applied to various lay-up optimization problems. In particular, branch and bound, genetic algorithms, particle swarm and ant colony have been used. Note, this list is neither exhaustive nor is it intended to be. The methods detailed here have received the most academic and industrial interest. As such, this chapter will focus on these methods alone. Initially, a branch and bound approach is considered. It is shown that whilst this approach is complete, that is, if a solution exists the algorithm will find one, the method is known to be computationally inefficient. This is pertinent when the number of plies and/or the number of possible ply orientations is large. Motivated by this, three meta-heuristic approaches are considered. These are a GA, PSO and ACO. Through formal analysis and a series of numerical examples, it will be shown that the ACO and PSO offer the most suitable route to determining laminate stacking sequences (which satisfy a set of structural constraints) in laminated composite design.

4.2 Background and Literature Review

Lay-up optimization of laminated composites has evolved significantly over the past 25 years. Ghiassi et al. (2009) recently provided a detailed review of the various optimization techniques which have been successfully applied to laminated composite design optimization. However, recent focus has been on the optimization of laminated composites using lamination parameters and/or meta-heuristic approaches. A two level optimization approach has recently been adopted by Herencia et al. (2008a) to solve the stacking sequence problem. At the first level, gradient based methods (SQP) were used to determine optimal lamination parameters and plate thicknesses. All constraints such as strength and buckling were embedded at this level and necessary trade-offs considered. As previously stated, lamination parameters are particularly useful intermediate design variables in the optimization of laminated composites because the constraining relationships between lamination parameters form a convex feasible region (Grenestedt and Gudmundson 1993). Consequently, where the objective function and constraints are a convex function (in minimization problems) of the design variables, gradient based methods guarantee that global optima are obtained. At the second level of the optimization Herencia et al. (2007a-c, 2008a) used a GA (ply orientations as discrete variables) to obtain a stacking sequence which satisfies the set of design constraints. To remind the reader, it is important to note that at the second level the fitness function is highly non-convex. The non-convexity of the fitness function arises due to the mapping between ply orientations, lamination parameters and the fitness function. As such, gradient based methods may only find local optima and thus not entirely appropriate. Motivated by this shortfall, meta-heuristic optimization methods have been proposed. The chief benefit of using a meta-heuristic algorithm is that gradient information is not required. Whilst local optima remain a problem, the ability to escape local optima can be studied and certain algorithm parameters can be adjusted to improve the performance of each method.

Lay-up optimization is inherently discrete. In laminate design, ply thickness is generally fixed and ply orientations take a range of discrete values. Determining a stacking sequence for a given plate thickness using ply orientations as design values is a combinatorial problem. For a laminate of n plies where each ply orientation can take a value in a set of size m, it follows that the number of possible designs is m^n. As such, enumeration quickly becomes increasingly difficult due to the combinatorial explosion of possible lay-ups with increasing thickness and an increasing set of ply orientations. To solve this problem, a number of techniques, both deterministic and probabilistic have been proposed. With respect to deterministic optimizers, a branch and bound method has been proposed, chiefly by Todoroki and Terada (2007) and Sekishiro and Todoroki (2008). The authors successfully implemented a branch and bound approach (where the laminate thickness is fixed) for a variety of problems using a single level optimization routine. In particular, they maximized the buckling load of a laminate with fixed thickness using a quadratic polynomial approximation of the buckling response function with a design of experiments (DOE) approach. Despite the complete/deterministic nature of a branch and bound method, the algorithm is known to be computationally expensive and potentially inefficient. Whilst Todoroki and Terada (2007) eliminate the former problem by approximating the objective function, the general formulation suggests that computational issues remain a problem. The efficiency issues concerning the branch and bound approach become apparent when the set of possible ply orientations or number of plies increases and the objective function and constraint (evaluations) are time intensive finite element computation calculations. Furthermore, if the laminate thickness is not fixed, it is feasible that the branch and bound approach may become even more inefficient. To overcome this problem, population based meta-heuristics have again been proposed. This is due to their ability to be configured and adjusted to suit the problem at hand. The success of GAs in composite optimization has been well documented. For example, Le Riche and Hatka (1997), Nagendra et al (1999), Liu et al (2001), and Herencia et al (2007a-c,2008a-b) have all successfully applied GAs, both directly and as part of a multi-level approach, to composite optimization. Moreover, GAs naturally lend themselves to discrete variables as each gene in a GA can represent a single angle in a lay-up. GAs are known as a robust approach and have seen commercial as well as academic success. Despite the popularity of GAs, they may be computationally less efficient than other meta-heuristic approaches and often require extensive parameter refinement to ensure reasonable convergence times. That is, the number of stacking sequences evaluated in order to find feasible solutions may be higher than other methods. In contrast, Particle Swarm Optimization (PSO), inspired by the notion of birds flocking (Kennedy and Eberhart (1996)) may require less computational effort and parameter refinement. Initially designed for continuous domains, the paradigm has been successfully applied to discrete problems such as lay-up optimization with minimal change in the original formulation. Suresh et al. (2005) further highlighted the benefits of a discrete PSO applied to a multi-objective composite box design and highlighted the gains in using a PSO over a GA. Kathiravan and Ganguli (2008) recently used a gradient based method and PSO to determine laminate stacking sequences to satisfy strength failure criteria in a box beam. The analysis highlighted that a PSO was able to determine a global optima, whereas a gradient based optimizer would only converge to local optima. Despite this shortcoming, the local optima were often good points and could be used as starting points in the PSO. Omkar et al. (2008) successfully applied a variant of the standard PSO vector evaluated particle swarm optimization (VEPSO) model introduced by Kennedy and Eberhart (1996). The authors solved a multiple objective problem of minimizing weight and the total cost (monetary) of the composite structure subject to three failure criteria: failure mechanism based failure criteria, maximum stress failure criteria and the Tsai–Wu failure criteria. Recently, the Ant Colony Optimization (ACO) paradigm, first proposed by Dorigo in 1993, has been receiving growing interest. An ACO simulates the behaviour of ants foraging for food. It is noted that the ACO paradigm is specifically designed for combinatorial problems such as stacking sequence optimization. This is in direct contrast to the PSO which was designed for continuous domains. Aymerich and Serra (2007, 2008) demonstrated the effectiveness of the ACO in maximizing the buckling load of a composite plate (with different boundary conditions) subject to strength constraints. The authors also showed significant efficiency gains over a GA. Aymerich and Serra (2008) demonstrated that an ant colony could rapidly determine stacking sequences and, furthermore, that this approach was robust and reliable. Naturally, the implementation of the aforementioned optimization algorithms can dramatically affect the performance of the algorithm. As such, the most appropriate implementation for each approach will be discussed.

In this chapter, after a brief discussion of the branch and bound approach outlined in the literature, a GA, PSO and ACO are analysed and compared in detail. The analysis aims to show two key points, a) how the number of possible ply orientations affects algorithm convergence and b) how does laminate thickness (complexity due to number of plies) affect convergence. Through an efficiency, reliability and robustness analysis, it will be shown that an ACO and PSO offer potentially the best routes to determining laminate stacking sequences. It is further concluded that the selection of the optimization technique and optimum population size is likely to be problem dependent.

4.3 Deterministic Methods

4.3.1 – Branch and Bound

Branch and Bound (BB) is an optimization technique used for finding solutions to problems mostly discrete and combinatorial in nature. Like all optimization problems, the driving principle is to find a minimum (or maximum) value of an objective function over a feasible domain. The underlying procedures in the algorithm are partition, sampling, and subsequent lower and upper bounding. These operations are applied iteratively to the set of active subsets within the feasible set. BBs guarantee that global optima are found due to their exhaustive search feature. However, the quality of solution can often come at a high computational cost.

The general schema for a BB is as follows. The first step is the effective division of the feasible region into smaller sets containing feasible solutions which are examined more closely using a local search. This local search, however, does not incorporate a different optimization algorithm. Rather, this is part of the BB procedure. The central idea to branch and bounding is that when moving through a tree structure, if the lower bound for a subregion A from the search tree is greater then the upper bound for any other previously visited sub-region B, then A may be safely discarded from the search. This step is called pruning. The pruning stops when all nodes of the search tree are either pruned or their minimal values are known. At this point, all non-pruned sub-regions will have their upper and lower bounds equal to the global minimum of the function. At this point, the algorithm terminates

It should be noted that the efficiency of the BB method is critically dependent on the choice of BB algorithm. At worst, the BB algorithm could lead to repeated branching without any pruning until the sub-regions become very small. In this case, the BB method is the same as enumerating the whole domain (exhaustive search). This is an impractical and complex task, especially in laminate composite design where the number of possible lay-ups maybe gargantuan. With respect to bounding, the algorithms used are problem specific and no generic algorithm exists. On a positive note, as the branching and bounding of different sub-regions can be computed independently, it follows that the BB naturally lends itself to parallel and distributed implementations.

Todoroki and Terada (2008) used a BB method for stacking sequence optimization. Lamination parameters were used as design variables, and thus the problem exists in lamination parameter space. The authors used a BB approach to maximize the buckling load of a laminated composite plate. Hence, the problem exists in the out-of-plane lamination parameter space. The authors restricted their study to $0, 90, \pm 45$ where it is noted that $\xi_4^D = 0$. Thus $(\xi_1^D, \xi_2^D, \xi_3^D)$ were used as design variables and the dimension of the search space was three. In order for the BB method to be applied to the stacking sequence optimization problem, the authors introduced several assumptions. Firstly, ply thickness was considered to be uniform. Secondly, the number of plies in the lay-up was known (similar to the second level of the optimization presented in this thesis) and finite. Consequently, there are only a finite (albeit large) number of possible stacking sequences. As previously stated, the first step of the BB algorithm is branching. By plotting all feasible lay-ups in the out-of-plane lamination parameter space it is observed that the feasible region is a sum of smaller sub-regions. This is clearly shown in Fig. 4.1.

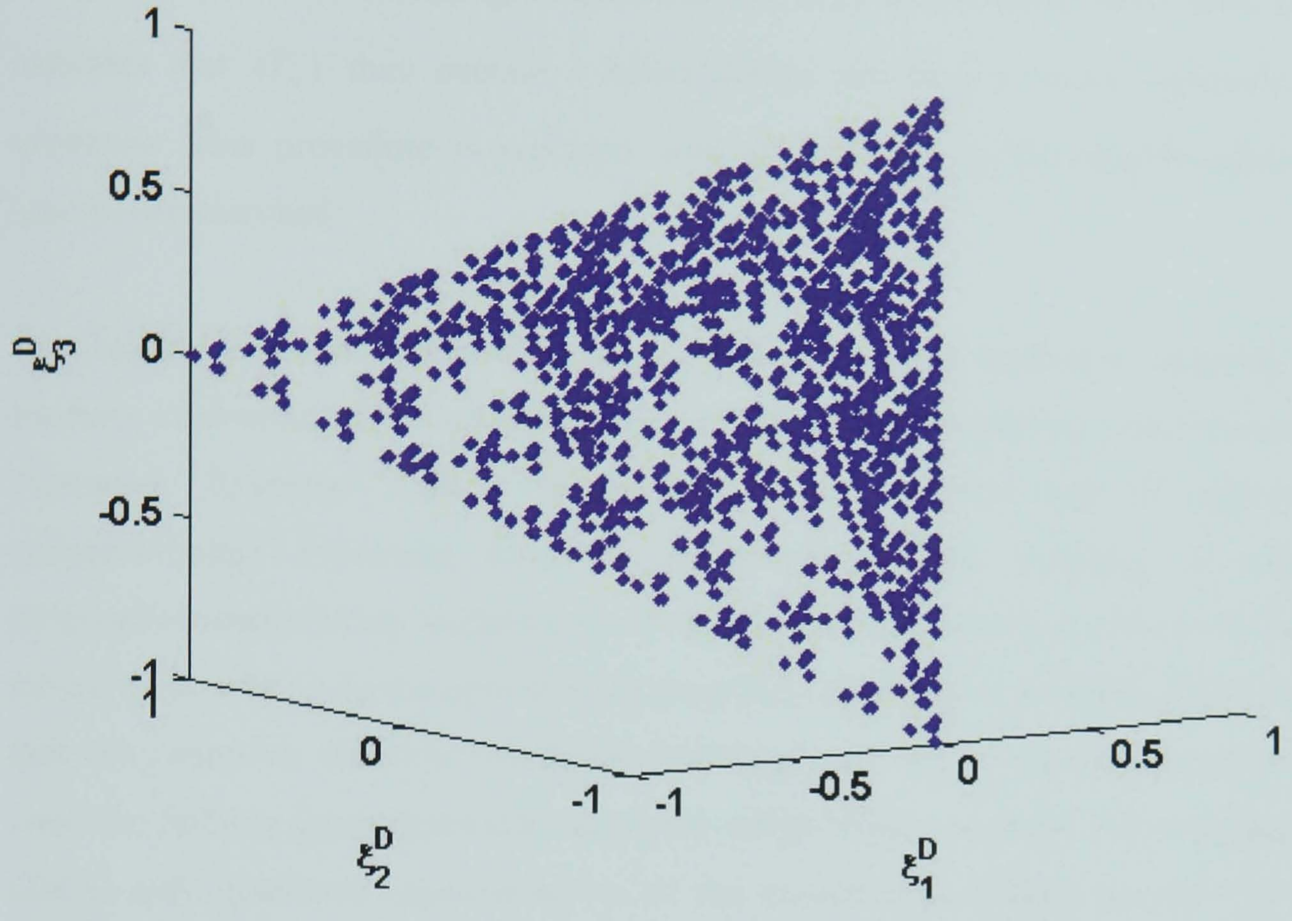

Fig. 4.1 – Fractal Image of the Feasible Region of $\xi_1^D, \xi_2^D, \xi_3^D$ for angles restricted $[0, 90, \pm 45]$

The second step of the BB algorithm is bounding. Observe from Fig. 4.1, the fractal pattern of feasible lay-ups forms a tetrahedron, *T*. To initiate the branching procedure, a starting point is required. Todoroki and Terada (2008) used the balanced symmetric lay-up $[45, -45, 45, -45]_s$. Note this lay-up forms a self similar tetrahedron (T_0) inside the tetrahedron shown in Fig. 4.1 For details see Todoroki and Terada (2008). The branch searching processing now begins. Suppose that the first ply angle searched is zero degrees. Thus for eight layer laminate, the branch search would be $[0, \theta_2, \theta_3, \theta_4]_s$. From the fractal patterns determined by plotting all feasible laminates, the stacking sequence $[0, \theta_2, \theta_3, \theta_4]_s$ corresponds to a self-similar tetrahedron inside the tetrahedron which is inside the tetrahedron shown in Fig. 4.1. As the BB algorithm moves from node to node, a new self-similar tetrahedron is formed. If the new tetrahedron (T_N) contains a lower value then the provisional one (T_0), then this implies that the new tetrahedron is less likely to contain a better design. Conversely, if (T_N) has a higher value then (T_0), this indicates that (T_N) may contain a better design and thus a 'more' optimal stacking sequence. This procedure is repeated until all branches in the neighbourhood (local search) are searched.

To evaluate each lay-up (node in the search), a response surface is required. This is because, evaluating each node using FEA would be time consuming and computationally expensive. Response surface methodology is an effective way of approximating computationally expensive functions via sampling. The sampling is used, with polynomial interpolation to determine a response surface which is effectively used as a closed form solution in the optimization procedure. Todoroki and Haftka (2005) observed that the objective function (maximize buckling load) in lamination parameters space could be suitably approximated by quadratic polynomials. As such, Todoroki and Terada (2008) used quadratic approximations of the objective (buckling) function to create a response surface in terms of lamination parameters. The coefficients of the polynomial were determined using the design of experiments (DOE) method. For more information see Todoroki and Terada (2008). Thus for a given objective function (where the approximation is assumed to have been determined), the objective is to find a stacking sequence that optimizes (maximizes or minimizes) the response. It is generally accepted that an efficient evaluation function is needed to 'prune' the branches of the tree and thus move towards an optimal solution more rapidly. By approximating the buckling function, Todoroki and Terada (2008) achieved this objective. Note, for any DOE approach, there exists the general issue of determining the number of observations required for statistical robustness whilst minimizing the associated computational cost of evaluating FEA models to obtain the response surface. Essentially, this creates an optimization problem within an optimization problem.

With respect to stacking sequence optimization using the BB method, it should be noted that as the number of possible ply orientations increases, so will the complexity of this combinatorial approach. Additionally, the number of plies in the lay-up also creates complexity. Therefore, a branch and bound can be considered quite inefficient However, for certain problems where the total number of layers is small, say where $n < 10$ and ply thickness is uniform, then the branch and bound method may be used to search for the optimum subject to design constraints. Whilst seemingly computationally inefficient, the underlying simplicity of the BB is striking and will be revisited in Chapter 5.

To overcome the issues relating to BB for thick laminates (or where number of ply orientations is large), meta-heuristics can be used. The use of meta-heuristics in optimization and particularly composite optimization has been extremely popular. Meta-heuristic techniques apply standard mathematical procedures with an underlying idea or heuristic to determine so called 'good solutions' to highly complex or computationally intensive problems. Whilst meta-heuristics are rarely used to find global optima, their ability to locate good solutions in multi-modal problems where gradient based optimizers fail, make them an attractive option. The meta-heuristics used in this thesis derive from swarm intelligence as well as population based evolution. Meta-Heuristics allow the designer to fine tune the algorithm to solve the particular problem at hand as well as implementing a knowledge base to solve these problems. In summary, a meta-heuristic approach provides a dynamic and powerful approach to solve complex combinatorial problems such as lay-up optimization. This thesis considers three key meta-heuristic optimization approaches: Ant Colony Optimization (ACO), Genetic Algorithm (GA) and Particle Swarm Optimization (PSO).

4.4 Meta-Heuristics

Meta-heuristics are computational techniques used to generally solve complex combinatorial problems. The meta-heuristics adopted in this thesis are swarm/population based techniques. Note, there also exists meta-heuristic optimization routines which are not population/swarm based such as Tabu search. Also, meta-heuristics are zero order methods and as such do not require the gradient of the function or constraints under consideration. In contrast, function values are used in conjunction with an underpinning heuristic rule to search for better solutions. Meta-Heuristics have been successfully applied to 'hard' optimization problems. Such examples include combinatorial problems including lay-up optimization. Meta-heuristic techniques generally contain the same format; generate an initial population, evaluate the population, change the population via a heuristic rule and repeat until the population or member of the population satisfies a stopping criteria. This scheme is demonstrated in Fig. 4.2.

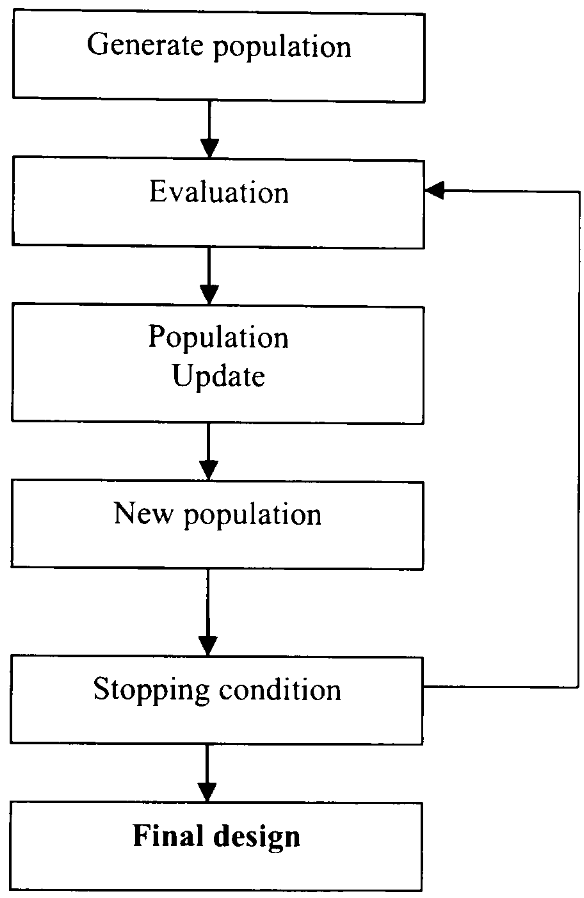

Fig. 4.2 – Meta-Heuristic Block Structure

Unlike gradient based methods, meta-heuristics do not require gradient information. As such, they do not suffer the immediate shortfall of locating only local optima. Whilst local optima remain a problem, the meta-heuristic can be studied and used to avoid local optima and search for global minima. Additionally, meta-heuristics can be applied to non-continuous and non-differentiable functions increasing the scope of their applicability. With respect to gradient based methods (see Chapter 3), most are first or second order approximations, e.g. a Taylor expansion of the Lagrangian function (again, see Chapter 3). Assuming the underlying problem has n variables, at each iteration of the gradient optimizer there is one function evaluation (single stacking sequence) plus n evaluations where forward finite differences (gradient approximations for each variable) are used, hence $n+1$ evaluations. Note, this holds if and only if a numerical optimization is used.

In contrast, if central finite differences are used, $2n + 1$ evaluation per iteration take place. For zeroth order methods, no gradient calculations are necessary. As such, one can consider the population size of a meta-heuristic to be the same as the number of evaluations at each iteration. Therefore, the population size can be at least $n+1$ depending upon the benchmarking method for gradient approximations. Herein, this is defined as *computational equivalence*. Whilst a gradient based method will determine local minima, population based zero*th* order methods can be used to escape local optima and find global solutions. Whilst it is recognized that global solutions are not always necessary in engineering, local optima is an issue which cannot be avoided in constraint satisfaction problems (CSP) such as detailed in this thesis.

Thus, it is feasible to compare gradient based methods with meta-heuristics in terms of computational equivalence. As such, maintaining a population size of $n + 1$ (which corresponds to the number of function evaluations using a forward finite difference approach) solutions gives greater scope to explore the search space. Next, GA, ACO and PSO approaches are detailed and a comparison is made thereafter.

4.4.1 Genetic Algorithms

In this thesis a traditional/standard GA is employed. That is, the GA is used to determine the number of each ply orientation as well as the stacking sequence. In contrast, a permutation GA determines an optimal stacking sequence given a fixed number of each ply orientation. In the GA, natural representation (no encoding) is adopted. Ply orientations act as chromosomes and hence a stacking sequence is represented by a string of chromosomes. A GA can be summarized in the following (pseudo-code) algorithm

1. Generate an initial population

2. Evaluate the fitness of each individual in the population using the fitness function F

3. Select an element of the population for reproduction

4. Breed new generation through crossover and mutation (genetic operations)

5. Repeat steps 2-4 until a member of the population satisfies the stopping condition (this is where the minimum reserve factor is greater or equal to one.

The genetic operators, crossover and mutation are used to breed a new generation based upon the fitness of the current population. Crossover relates to the permutative action of the GA and is demonstrated in Figs. 4.3a and 4.3b.

60	60	45	-45	45	45	-45	45	60	60

60	-60	-45	-30	90	90	-30	-45	-60	60

Fig. 4.3a - Parent One and Two

60	-60	45	-45	45	45	-45	45	-60	60

60	60	-45	-30	90	90	-30	-45	60	60

Fig. 4.3b - Example Children from Parent one and two (Crossover)

In this instance, a ply will swap positions with another ply in another stacking sequence, thus creating two new stacking sequences. The mutation operator allows the algorithm to avoid local minima by preventing the population of chromosomes from becoming too similar to each other (preventing a homogeneous population), thus slowing or even stopping evolution. With mutation, a string of chromosomes is selected and concatenated into another stacking sequence (string of chromosomes). This action is demonstrated in Fig. 4.4.

60	60	45	-45	45	45	-45	45	60	60

60	60	45	-30	90	90	-30	45	60	60

Fig. 4.4 - Example Child formed via Mutation

To prevent a homogeneous population, it is often useful to select a mixture of the fittest members of the population with some of the less fit members. This is in direct contrast to a ranking system or stochastic uniform approach and gives greater diversity to the population. It is noted that whilst this policy may limit short term convergence, it gives greater exploration of the search space and thus increases the likelihood of ultimately finding good designs. The implications of the population size and probability of crossover and mutation are discussed in the comparison section. Next, the ACO is formally introduced and discussed.

4.4.2 Ant Colony Optimization

An ACO (Dorigo 1992) contains a colony of ants, where each ant represents a notional traveler that traverses each step in a path. This journey is analogous to an ant walking and traversing each ply in a lay-up (from the bottom to the middle ply). Ants move through a multidimensional search space where the dimension of the search space equals the number of plies. At each iteration, an ant selects a path (lay-up) which changes according to its own experience and also the experience of the other ants in the colony. Initially, the paths of the k ants are randomly generated. Each ant represents a lay-up which is encoded by a route taken by that ant through the search space. At the end of the path, the path or lay-up, is evaluated using the fitness function F defined in Chaper 3. Each ant then deposits pheromone on each arc (ply in a given position) that it has visited. The amount of pheromone deposited is calculated as follows: if the ith angle is used as the jth ply in the kth lay-up,

$$\tau_{ij}^{k} = \frac{1}{F_k} \tag{4.1}$$

else

$$\tau_{ij}^{k} = 0 \tag{4.2}$$

where τ_{ij}^{k} is the pheromone matrix for the kth ant, $i = 1 .. \Phi$, where Φ is the number of angles in the design envelope and $j = 1 ... r$ where r is the half-number of plies for a symmetric laminate. The definition of the pheromone matrix given in Eqns. (4.1-4.2) ensures that a low fitness equates to a higher pheromone deposit and vice versa. Note, τ_{ij}^{k} cannot be infinite since the stopping condition is applied before evaluation. Note, if the value of the fitness function is zero, the algorithm is terminated and a solution is returned since this implies all design constraints have reserve factors greater or equal to one. The next step in the algorithm is the calculation of the ant routing table, ψ_{ij}^{k}. Let,

$$a_{ij} = \sum_{i=1}^{k} \tau_{ij}^{k} \tag{4.3}$$

then,

$$\psi_{ij}^{k} = a_{ij} \qquad \text{if} \qquad a_{ij} > 0 \tag{4.4}$$

else

$$\psi_{ij}^{k} = \max(a_{ij}) \tag{4.5}$$

for all angles i at a given position j

When an ant begins its path, it decides which ply to select using the information contained in Eqns. (4.4-4.5). Each ant then walks along a new path (lay-up). At each ply, the ply orientation is selected using the following probability function,

$$p_{ij} = \frac{\psi_{ij}}{\sum_{i=1}^{\Phi} \psi_{ij}} \quad \textbf{(4.6)}$$

Each ply angle i at position j in each lay-up is randomly selected with a probability p_{ij} which is calculated using Eqn. (4.6). In practice, the process of selecting ply orientations with given probabilities is achieved in MATLAB (2009) using the 'randsrc' function. It is noted that the process of selecting ply orientations is stochastic and thus having a higher probability does not guarantee selection. At the next iteration, pheromone concentration is then updated (for good designs) using the following function,

$$\tau_{ij}^{k}(t) \leftarrow \tau_{ij}^{k}(t) + \tau_{ij}^{k}(t-1) \text{ if } F_k(t) < F_k(t-1) \quad \textbf{(4.7)}$$

else,

$$\tau_{ij}^{k}(t) = \tau_{ij}^{k}(t-1) \quad \textbf{(4.8)}$$

where t is the tth iteration. It is noted that the pheromone matrix is only updated when the solution in the tth iteration is superior to the t-1 solution. Hence, the best lay-up that the ith ant has explored is used to update the pheromone. Note, in the ACO detailed herein, pheromone evaporation and heuristic information are not used. The ACO is therefore simplified in comparison to the algorithm detailed by Aymerich and Serra (2008) and Dorigo (1993). Furthermore, the ACO performed well (demonstrated in the numerical examples in this chapter) without the use of heuristics and parameter refinement. Next, the PSO implementation is discussed.

4.4.3 Particle Swarm Optimization

The particle swarm paradigm contains a swarm of particles (lay-ups), where each particle represents a potential solution to the optimization problem. Particles move through a multidimensional search space where the position of the particle is adjusted according to its own experience and also the experience of the other particles in the swarm. Let $x_{ij}(t)$ denote the jth ply (from the outer surface inwards) in the ith lay-up at the tth iteration. Note, t is a positive integer. The position of the particle (lay-up) is updated by adding a velocity vector, $v_{ij}(t)$ to the current position, i.e.

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1)\Delta t \tag{4.8}$$

Where,

$$v_{ij}(t+1) = \omega \cdot v_{ij}(t) + \frac{c_1 r_1(t)\cdot(L_{ij}(t) - x_{ij}(t)) + c_2 r_2(t)\cdot(G_j(t) - x_{ij}(t))}{\Delta t} \tag{4.9}$$

where c_i, ω are user-defined constants used to accelerate the swarm and control the 'inertia' of the swarm, respectively and $\Delta t = 1$. Functions $r_1(t)$, $r_2(t)$ are random numbers generated within the interval [0, 1], $L_{ij}(t)$ is the position (local best) of the jth ply in the i th lay-up in its history over all iterations,

$$L_{ij}(t) = \begin{cases} L_{ij}(t-1) & \text{if} \quad F(x_{ij}(t)) \geq F(L_{ij}(t-1)) \\ x_{ij}(t) & \quad\quad F(x_{ij}(t)) < F(L_{ij}(t-1) \end{cases} \tag{4.10}$$

where f is the fitness function which is to be minimized (see Chapter 3). The global best lay-up, denoted by G is calculated as,

$$G(t) = \arg\min\{F(L_i(t))\} \quad \forall i, t \tag{4.11}$$

By updating the position vector with the velocity vector over a small time period (iteration), Eqn. (4.9), each lay-up is either constrained to stay within its own neighborhood or is 'forced' to a different area of the search space. Note, the velocity vector $v_{ij}(t)$ drives the optimization and thus the convergence of the swarm to an optimal solution. The velocity vector $v_{ij}(t)$ also represents the change in ply angle of the jth ply in the ith lay-up in dimension j = 1...m/2 (or (m+1)/2 if mid-plane symmetric) where m/2 is the dimension of the search space (i.e. half the number of plies). Note, $r_i(t)$ introduces a stochastic element into the algorithm, to prevent premature convergence to local optima whilst for small values of ω, the swarm conducts a local search. On the other hand, if the value of ω is large, the swarm has enough impetus to explore the wider search space. By updating the position vector with the velocity vector over a small time period (iteration) using Eqn. (4.9), each lay-up is either constrained to stay within its own neighborhood or is 'forced' to a different area of the search space.

4.4.4 Comparison Between Meta-Heuristics

In the previous sections, the GA, ACO and PSO algorithms were discussed. Next, the three optimization techniques are compared. Interestingly, all three algorithms follow the same approach. That is, evaluate a population with a fitness function, change the population and re-evaluate the population. This procedure continues until the fitness of a member of the population satisfies a given stopping criteria. The key differences between the three algorithms concern how the population changes as well as the number of user defined parameters and their associated sensitivity. As such, this comparison will focus on these features. It should be noted that the GA and ACO are discrete optimizers. In contrast, the PSO is a continuous optimizer and therefore its results are rounded to achieve discrete values.

Firstly, we shall consider the GA. With a GA, a set of potential solutions (genes) evolve using the genetic operators crossover and mutation. Mutation and crossover are stochastic operators chosen to create new, and hopefully better, members of the population as well as diversifying the population. Selecting the appropriate values for the probabilities of crossover or mutation is a non-trivial task. The convergence of the GA is highly dependent upon the correct choice of these parameters. As such, a significant amount of time is required to fine tune these values for each problem. Moreover, different parameters may be more appropriate depending upon the dimension of the problem; in this case, the dimension corresponds to the number of plies. As such, it is suggested that optimal probabilities of mutation and crossover may be related to the dimension of the problem or indeed dynamically change during the optimization. However, investigations concerning a dynamic update are outside the scope of this thesis.

Whilst the permutative nature of the GA appears to be well aligned to solving the stacking sequence problem (note this thesis uses a standard GA), it is important to note several points. Firstly, if the correct number of each ply orientation could be selected, determining the stacking sequences is simply a case of permutating the stacking sequence. However, determining the number of each ply orientation (which is non-unique) is a highly non-trivial and computationally expensive task. To overcome this problem, a diverse population is required to breed the next generation of children (lay-ups). Secondly, maintaining a diverse population is not easily achieved. It is further noted that the convergence of a GA (like other heuristic optimizers) is heavily reliant upon the initial population. A bad initial population can slow down convergence or prevent convergence altogether. For the sake of efficiency, a pseudo-random initial population is selected. At this juncture, it is important to clarify the meaning of a pseudo-random population. A pseudo-random population is a set of lay-ups (given size) pseudo-randomly generated from a set of possible ply orientations.

In many problems, GAs may have a tendency to converge towards local optima. The likelihood of this occurring depends upon the shape of the response surface as well as the algorithm's ability to maintain a diverse population. It is important to note that the response surface referred to herein is highly non-convex and thus the potential of finding local minima is relatively high (although no gradient information is required). The non-convex response surface (fitness function) arises due to the trigonometric mapping between ply orientations and lamination parameters. This is further complicated by the non-linear mapping between lamination parameters and the fitness function. To overcome the problem of premature convergence, several solutions are proposed. Firstly, a niche penalty (Nagendra et al. 1993) can be introduced if the mean of the population's fitness approximates the best fitness. In this instance, a proportion of the population is replaced with pseudo-randomly generated stacking sequences. Additionally, a dynamic rate of mutation may be more appropriate. Note, in this thesis, and for the sake of consistency, a niche penalty is introduced to all three algorithms.

Unlike a GA, the PSO contains no genetic operators such as mutation and crossover. Particles are updated relying on the fact that each particle within the swarm has a memory. Compared with a GA, the information sharing mechanism in a PSO is significantly different. In a GA, chromosomes share information with each other. In contrast, a PSO, only shares the global best position. Therefore, a PSO has a one-way information sharing mechanism. In contrast, a GA has a two-way sharing mechanism since the genes of each member may be ultimately shared with each other. In this sense, the members are not necessarily following the best positions; they simply swap in order to improve the design. Despite the absence of genetic operators, the PSO has three control parameters: c_1, c_2, ω. The PSO parameters c_1 and c_2 impact the convergence rate of a swarm. The inappropriate choice of these parameters can slow down convergence or may induce cyclic behaviour. As such, it is important to find appropriate values for these parameters, which may be problem specific. A trial-and-error exercise was undertaken and it was found that a choice of $c_1 = c_2 = 1.5$ yields good convergence characteristics and furthermore, solutions were found for these values. With reference to Eqn. (4.9), assigning $c_1 = c_2$ is somewhat intuitive since if $c_1 > c_2$ the swarm may converge towards a local best and thus local optima. Conversely, if $c_1 > c_2$ the swarm may prematurely converge towards the swarm global best. The parameter ω controls the inertia of the small. If ω is high, the swarm conducts a diverse search. On the other hand, small values for ω imply a localized search. Similarly to a GA, the PSO may require different control parameters depending upon the dimension of the problems. As such, dynamic control parameters may be more appropriate but again is outside the scope of this thesis. One particular issue that was found with the PSO was velocity scaling. When using natural representation (no encoding) with the PSO, the velocities need to be large enough to give momentum to particles to change. This is particularly evident with ply orientations as design variables, and with low value control variables. As such, the control variables may be need to be scaled to give greater impetus to the swarm and hence diversity.

Interestingly, despite the number of control parameters in the standard PSO, it was found that greater fine tuning was required for the genetic operators (in GA) as the thickness (dimension) changed, in comparison to the PSO parameters. Moreover, despite the number of parameters being higher in the PSO, this induced little effect. It is observed that the PSO is treated as a continuous optimizer (inline with the PSO paradigm) and after the velocity update, each ply position is rounded to the nearest position in the set of possible ply orientations – as such, the sensitivity between updated positions and rounded positions may rapidly change the search direction of the swarm (for better or for worse). Clearly, this problem is eliminated when a continuous or near continuous set of ply orientations are used.

In contrast, the ACO works quite differently. Each ant traverses a stacking sequence and selects a ply based on a probability. The probability relates to how much pheromone (how good a solution is) has been deposited on that particular ply in a given lay-up. If a certain ply has not been visited yet, the ant will treat that ply with the same probability as the current best orientation. In the Combinatorial Particle Swarm Optimization (CPSO) introduced by Yang et al. (2004), the velocity is used to determine the swap probability. The velocity is a weighted average of the global and local best positions. As such, the ACO represents an extension of the CPSO. This is because in the CPSO the swap is limited to the global best position. The continuous experience of the ants adjusts the probabilities accordingly. It is important to note that pheromone deposits in the ACO are not analogous to the velocity function of the standard PSO. This is because pheromone is used to determine the probability for each orientation at each position. In contrast, the velocity of the PSO is simply a measure of change. The information sharing mechanism of the ACO is different from the PSO and GA. Each ant shares its experience will every other ant. When each ant explores a new path, it makes a stochastic decision regarding which ply (continuously changing) to select based upon the experience of the other ants in the colony. It is noted, however, that if the set of possible ply orientations is large then the ACO, presented in the literature (Dorigo 1993, Aymerich and Serra 2008), may be slow due to the number of possible choices that each ant has to make. In contrast, a PSO provides only the information relating to the global best. A GA does not share all information, since only the chromosomes from the parents are used to form the next generation of children Whilst there is no function in the ACO which prevents premature convergence, the fact that each ply is probabilistically selected instead of being a weighted average of the local and global best increases the likelihood that a more diverse population is maintained and additionally good solutions are found. The ACO presented in the literature (Dorigo 1993 and Aymerich and Serra 2008) has several controllable parameters; however, it was found that no parameter refinement or pheromone evaporation was required in the present study. As such, this approach may offer a simpler method for determining laminate stacking sequences. In the current work, it was found that there was little need for pheromone reduction. The lack of parameter refinement is in stark contrast to the GA and PSO. Note, both the ACO and GA are naturally discrete optimization approaches. In contrast, the PSO is a naturally continuous optimization. In order to use the PSO with discrtee variables, rounding to the nearest design variable value is employed.

It is argued herein that the ACO may be the most appropriate method for determining stacking sequences. The benefits of the ACO are clear; little parameter refinement and, moreover, each ply is probabilistically selected based upon the experience of the other ants. On the other hand, PSO offers an attractive option when the set of possible ply orientations is large - since this maximizes the benefits (and continuous nature) of the PSO. The negative aspect of the PSO concerns the fine tuning of the control parameters; c_1 and c_2. The GA is appealing for the combinatorial nature of the stacking sequence problem, yet parameter refinement remains a significant problem. To ensure convergence in all three methods, the population must remain diverse. This need arises because a homogeneous population will become static and not converge. It should be noted, that several variations of each method detailed in this Chapter have been presented in the literature. It has been noted several times that the performance (convergence) of all three algorithms may be improved with the correct selection and size of the initial population. However, selecting a diverse or set of good designs is a non-trivial task (especially for hard combinatorial problems). As such, it would seem that any method that determines a good initial population will reduce the relative efficiency of the optimization routine. Therefore a pseudo-random population is generated from the set of ply orientations

4.5 Implementation

With respect to the GA, the selected implementation was that detailed by Herencia et al. (2008a) and implemented in MATLAB (2009a). The selected PSO implementation is taken from Kennedy and Eberhart (1996). The implemented ACO is the same as first presented by Dorigo (2003) and used by Aymerich and Serra (2008) yet without the use of heuristic information. The values of the parameters for the GA, ACO and PSO are detailed in Table 4.1.

Table 4.1 – Algorithm Implementation Details

Method	Population Size	Parameter One	Parameter Two	Parameter 3
GA	20	0.9 (Crossover)	0.25 (Mutation)	N/A
PSO	20	$\omega = 0.9$	$c_1 = 1.5$	$c_2 = 1.5$
ACO	20	$\alpha = 1$	$\beta = 1$	$\rho = 1$

Concerning the population size, as mentioned, this was fixed at twenty. This value may be optimal or sub-optimal not only for each of the three algorithms (GA, ACO, PSO) but moreover for each individual problem. As the population size increases, greater diversity is obtained and hence the algorithms may become more reliable. On the other hand, as the population size increases, the overall efficiency of the algorithm decreases. As such, there exists a complex trade-off between the population size and the response of the optimization. In order to compare the optimization methods on a like-for-like basis, a population size of twenty was selected and is a common value found in the literature (Kennedy and Eberhart 2006, Aymerich and Serra 2008).

4.6 Numerical Examples

In this section, a series of numerical examples are provided. Firstly, three different sets of ply orientations are considered,

i) $[0, 90, \pm 45, \pm 30, \pm 60]$ degrees

ii) [−90,90] in 15 degree increments

iii) [−90,90] in 5 degree increments

For each case study, three structural problems are used – each being a simply supported rectangular composite plate of dimensions 1200mm x 200mm. The following three load cases are considered,

1) $Nx = 0, Nxy = -700$ N/mm
2) $Nx = -800, Nxy = 0$ N/mm
3) $Nx = -600, Nxy = 300$ N/mm

It is noted that a) the loads are representative of the typical minimum and maximum compression and shear values for a plate in an aircraft structure and b) all aspect ratios are sufficiently large to allow use of the buckling closed form solutions detailed in Chapter 3. Please note, transverse shear deformation effects are not considered. By using different loading conditions, laminates of various thicknesses are generated. Such thicknesses (and number of plies) can then be used to evaluate the effectiveness of the optimization algorithms (in conjunction with the various sets of ply orientations). In the optimization, no restrictions are placed upon the stacking sequences. For example, there is no balanced laminate requirement; four ply rule or 10% percent rule as outlined in Chapter 3. However, the integration of such constraints is straightforward and demonstrated in Herencia et al. (2008a). Next, the material properties used in the optimization are shown in Table 4.2.

Table 4.2 - Material Properties for Carbon/Epoxy

Material	E_{11} (GPa)	E_{22} (GPa)	G_{12} (GPa)	ρ (kg/mm^3)	ν_{12}
Carbon Epoxy	130	7	4	1.6e-6	0.3

Using the feasible region corresponding to $0, 90, \pm 45, \pm 30, \pm 60$ degree plies, calculated using the method detailed in Chapter 2, the continuous optimum lamination parameters and thicknesses for each load case are obtained and detailed in Table 4.3. Once the minimum thickness is established, the total number of plies in the laminate is fictitiously changed using three different ply thicknesses; 0.25mm, 0.125mm and 0.063 mm. For each loading case and number of plies, three different sets of ply orientations are considered. In the numerical examples, the following attributes are considered: reliability (total number of successes in finding a solution divided by total number of runs), robustness (range of obtained reserve factors) and efficiency (average number of iterations taken to determine a solution). Additionally, for each sub-problem the optimization algorithms were run 100 times to obtained averaged results. This was deemed sufficient in order to assess the degree of randomness in each meta-heuristic technique. The maximum number of iterations for each method was 200. The stopping criterion is if the laminate had reserve factors of greater or equal to one for the structural constraints. The performance of each algorithm is shown in Tables 4.4-4.6. Note example lay-ups are detailed in Table 4.7 for $0, 90, \pm 45, \pm 30, \pm 60$. In assessing Tables 4.4-4.6, several general observations are made:

1. Reliability and robustness of the GA decreases as the set of possible ply orientations increases.
2. ACO is the least sensitive to increases in number of plies.
3. PSO performance improves as the number of possible ply orientations increases.
4. For inherently discrete sets of possible plies, the GA is very reliable.
5. If the set of ply orientations is large or near continuous, PSO may offer the best route
6. Difficult problems will require a significant number of iterations to determine a solution.

Table 4.3 – Optimum Plate Thickness and Lamination Parameters

Load Case	***t* (Continuous)**	ξ_1^A	ξ_2^A	ξ_3^A	ξ_4^A	ξ_1^D	ξ_2^D	ξ_3^D	ξ_4^D
1	3.91	-0.351	-0.640	0.290	-0.349	-0.448	-0.527	0.788	-0.810
2	5.22	0.296	-0.541	0	0	0.024	-0.976	0	0
3	4.95	0.061	-0.660	-0.531	0.043	-0.027	-0.966	-0.124	0.045

Table 4.4 – Reliability Comparison (%)

Number of Plies (T=2*NP)	Load Case	GA Ply Set i)	GA Ply Set ii)	GA Ply Set iii)	PSO Ply Set i)	PSO Ply Set ii)	PSO Ply Set iii)	ACO Ply Set i)	ACO Ply Set ii)	ACO Ply Set iii)
8	1	98	98	99	100	100	100	100	100	100
16	1	99	96	93	98	99	100	100	100	100
32	1	89	91	85	98	99	100	100	100	100
10	2	99	100	99	100	99	100	100	100	100
21	2	99	98	93	100	99	100	100	100	100
42	2	87	88	88	100	100	100	100	100	99
10	3	100	98	93	99	100	100	100	100	100
20	3	99	94	95	99	100	99	100	100	100
30	3	96	97	96	100	100	98	100	100	99

Table 4.5 – Robustness Comparison (Average fitness – Minimum Constraint RF)

Number of Plies	Load Case	GA Ply Set i)	GA Ply Set ii)	GA Ply Set iii)	PSO Ply Set i)	PSO Ply Set ii)	PSO Ply Set iii)	ACO Ply Set i)	ACO Ply Set ii)	ACO Ply Set iii)
8	1	1.01	1	1	1.01	1	1	1	1	1.01
16	1	1	1.01	0.97	1	1	1.02	1.02	1	1.02
32	1	1.01	1	0.98	1	0.99	1	1.01	1.03	1
10	2	1	0.98	1	1	1	1	1	1	1
21	2	0.97	1	1	1	1	1.01	1	1	1.01
42	2	1	1	0.99	1.03	1	1	1.02	1.01	1.01
10	3	1	0.99	1	1	0.99	1	1	1.01	1
20	3	0.08	1	0.98	1	1	1	1.01	1	1.02
30	3	1	1.01	0.99	1	1.02	1	1	1	1

Table 4.6 – Efficiency Comparison (Average Number of Iterations)

Number of Plies	Load Case	GA Ply Set i)	GA Ply Set ii)	GA Ply Set iii)	PSO Ply Set i)	PSO Ply Set ii)	PSO Ply Set iii)	ACO Ply Set i)	ACO Ply Set ii)	ACO Ply Set iii)
8	1	31	44	61	15	21	14	8	10	11
16	1	45	67	87	23	24	21	10	10	13
32	1	59	98	103	31	28	27	12	15	23
10	2	21	35	41	14	17	15	4	11	13
21	2	41	50	57	15	21	14	7	8	11
42	2	61	87	115	23	24	21	13	16	23
10	3	31	57	82	31	28	27	8	9	14
20	3	36	45	61	14	17	15	10	11	13
30	3	41	52	76	12	17	21	12	17	21

Table 4.7- Example Stacking Sequences for Load Case (i)

Method	Lay-Up	Minimum RF
GA	$[60_2/45/60_4/-60/-45/60_2/-45/60/-60/45/-45]_S$	1
PSO	$[60_5/-60/60_2/-30/45/-60/60/-45/-60/45_2]_S$	1
ACO	$[60_3/45/60_2/45/-45/60_2/-60_2/45]/-30/-45_2]_S$	1

Table 4.8- Example Stacking Sequences for Load Case (ii)

Method	Lay-Up	Minimum RF
GA	$[-45/45_2/-45/45/-30/45/-45/-30/-45_3/0/30/-30_2/0/30/0]_S$	1
PSO	$[-45/45_2/-45/45_2/-45/45/-60/-45_2/30/-30/0/45/0/-30/90/$ $0/90/0]_S$	1
ACO	$[-45/45_3/-45/30/-45/45/-45/-60/60/-45/45/-30/45/30/0_4/-30]_S$	1

Table 4.9- Example Stacking Sequences for Load Case (iii)

Method	Lay-Up	Minimum RF
GA	$[-45/45/60/45_2/-45/-60_2/-45/-60/45/-30_2/-60/0/90/0/60_2/-45]_S$	1
PSO	$[45/-45_3/45/60/-45/-60/-45/30_2/-60/-30/90/0/-60_2/-30/0/90]_S$	1
ACO	$[45/-60/-45_2/45_2/-45/60/-45/45/-45/-30/30/90/-30/-60/-45/-30/0_2]_S$	1.1

It can be seen from Tables 4.3-4.9, that the GA was able to determine stacking sequences for most sub-problems. The GA had an average reliability of 98% which clearly decreased as the set of possible ply orientations increased. It is noted that when a solution was found, the algorithm was relatively efficient. The average number of iterations (Table 4.5) suggests that the GA's convergence is related to the dimension and hence the number of layers in the stacking sequence. This further supports the claim that the genetic operators are influenced by the dimension of the problem. It is argued that a dynamic rate of mutation of crossover may give better convergence rates. The GA often converged prematurely hence the total number of no solutions found. This may imply that appropriate values for genetic operators were not selected or that the population was too homogeneous. In essence, this would imply that the GA was unable to escape from a 'bad path'.

From Table 4.4 it can be seen that the ant colony optimizer performed extremely well with reliability of 100%. It is observed that as the number of possible ply orientations increased, so did the average number of iterations, yet the algorithm remained reliable and robust (shown in Tables 4.4-4.6). This shows that as the problem becomes increasingly continuous in nature, the number of iterations increases. This feature is an expected result, since the original ACO was designed for hard combinatorial problems. However, despite the increase in the number of iterations, the ant colony found a solution every time. Furthermore, the ACO was robust since the performance showed little variation as the number of plies changed (due to the complexity of the problem)

From Table 4.5, it can be seen that a PSO offers an efficient approach to determining laminate stacking sequences. Furthermore, the PSO is able to determine high quality solutions demonstrated by the mean best fitness (Table 4.6). The performance of the PSO increased as the number of possible ply orientations increased. This would suggest that the PSO may indeed be a suitable as both a discrete and continuous optimizer. However, since the ACO outperformed the PSO for inherently discrete sets of ply orientations, it may be more appropriate to select an optimization routine based upon the complexity of the problem. It was observed that the convergence of the PSO was highly related to the dimension of the problem. That, is to say, the thickness (number of plies) influenced the rate of convergence. In section 4.4 it was noted that the PSO parameters may require a degree of tuning depending on the problem. Additionally, it was found that the PSO also offers a highly robust approach to determining laminate stacking sequences. Lastly, the average reliability, shown in Table 4.3, of the algorithm was found to be close to 100%.

In the analysis, the emphasis is on the likelihood of determining a stacking sequence rather than the actual stacking sequence itself. However, in Tables 4.7-4.9, example stacking sequences are detailed for all three loading conditions with $0, 90, \pm 45, \pm 30, \pm 60$ degree plies. The variance in obtained stacking sequences can be partly explained by the random nature of each algorithm. Furthermore, due to the non-bijective mapping between lamination parameters and stacking sequences, there exist many possible stacking sequences for each set of lamination parameters. Lastly, it is noted that the constraints influence the complexity of the problem and the resulting fitness function. Additionally, different objectives (or constraints) such as the four ply rule (to avoid large matrix cracking) can be imbedded in the optimization techniques and would yield different stacking sequences. With respect to the four ply rule, it is interesting to note for the GA and PSO a penalty function would be required when including such a constraint. In contrast, the ACO would not require this, since when the ant traverses its new path if four have the same plies have been subsequently visited, that particular ply orientation would be excluded from the fifth ply. In summary, the four ply rule may reduce the efficiency of the PSO and GA, but potentially not overly affect the ACO.

4.7 Conclusions

The purpose of this chapter was the analysis and benchmarking of optimization techniques for lay-up optimization. Initially, a branch and bound method was considered. Whilst laudable, the issues concerning computational complexity outweigh the deterministic benefits of such an approach. In response, the analysis of three meta-heuristic optimization techniques to determine composite lay-ups was presented. Each of the three techniques were discussed and compared. The analysis suggested that the ACO paradigm would be the most appropriate method to determine laminate stacking sequences. Benchmarking was undertaken for a mass minimization problem subject to strength and buckling constraints. Later, a reliability analysis was performed. The results showed that the ACO is indeed the most appropriate method. The results indicated that the ACO was generally the most robust and reliable method given varying thickness (number of plies) and possible ply orientations. However, the ACO was not always the most efficient of methods - this was apparent when the size of the possible sets of ply orientations was near continuous. Nonetheless, the ACO was able to determine solutions in an acceptable number of iterations. When the number of possible ply orientations increased, it was demonstrated that the PSO outperformed the ACO. This effect is to be expected because the effectiveness of the ACO was diminished in running a near continuous optimization. However, if the set of possible ply orientations is close to or is indeed continuous, the ACO can be modified. This was recently shown by Socha and Dorigo (2008). For inherently discrete problems, the GA performed well with a high reliability factor but convergence was notably slow. On average, the GA was also the slowest of all three algorithms. In summary, it can be seen that the three different methods considered may be most suited to different types of problems. Motivated by the analysis and results of this chapter, the next chapter presents three modified approaches to stacking sequence optimization. These include a modified PSO, an adapted ACO and the introduction of stochastic discrete gradient optimizer. These developments consider the trade-off between reliability, efficiency and population size.

Chapter 5
Enhanced Second Level Optimization

5.1 Background

In Chapter 4, one deterministic and three meta-heuristic optimization techniques were introduced. Namely, a BB, GA, PSO and ACO were presented respectively. After a detailed analysis and a series of numerical examples, it was shown that an ACO and PSO offer the best route to solving the second level optimization problem. That is, the identification of a stacking sequence which satisfies the set of design constraints. The ACO was shown to give significant functionality and efficiency gains over a standard GA where the set of ply orientations was inherently discrete. In contrast, the PSO outperformed the GA chiefly when the set of ply orientations approached a near continuous set. Despite the success of the ACO and PSO over the GA, several modifications are proposed to enhance the efficiency and effectiveness of the two aforementioned approaches. As such, the objective of Chapter 5 is to detail these enhancements and demonstrate their effectiveness.

In this chapter, a modified particle swarm (MPSO) and a combined ant colony - direct branching method (ACO-DBM) are presented. Motivated by the DBM, a form of local search, a stochastic discrete gradient descent (SDGD) approach is also presented. It is shown that an ACO-DBM offers the best approach to lay-up optimization (at the second level) based on these modifications. However, the MPSO is shown to have distinct benefits, in particular where ply orientations can take continuous values. Additionally, the SDGD offers an alternative approach to identifying laminate stacking sequences in composite optimization.

5.2 Literature Review

The literature review for the current chapter is a slight extension to that presented in Chapter 4. With respect to the current chapter, it is important to identify and discuss literature relating to the inclusion of local search in a meta-heuristic approach. The idea of local search is not a new one and has been popular in the optimization of laminated composites. The use of local search is inspired by the notion that both meta-heuristics are able to determine good solutions relatively quickly, but local refinement (or local neighbourhood search) could potentially improve the designs. In the case of constraint satisfaction problems, local search in addition to a meta-heuristic may in-fact generate a feasible solutions thus solving the CSP. Note, a CSP is solved by determining a single feasible solution.

As previously discussed, Ghiassi et al. (2009) provided an extensive review of optimization techniques applied to laminated composite design. Ghiassi et al. (2009) highlighted the significant use of local search techniques. In particular, a branch and bound and Tabu search were discussed. However, it is interesting to observe that the ACO presented by Dorigo (1993) and Aymerich and Serra (2007, 2008) incorporates a pseudo-Tabu search. Recall, from Chapter 4 and Eqns. (4.3)-(4.5) the amount of pheromone deposited on a node is calculated based upon a Tabu table which contains information on whether a given ply orientation for that ply position has been visited during the path of a given ant. The BB approach, as outlined in Chapter 4, is also another form of local search. Todoroki et al. (2007, 2008) have pioneered the use of branch and bound in composite optimization. For further details, please see Chapter 4. Additionally, Kim and Hwang (2003) proposed a branch and bound method using the concept of an idealized laminate. An idealized laminate is one with physically unachievable stiffness properties but used to maximize or minimize an objective. The idealized stacking sequence (each layer having the best possible stiffness values) was used in the branch and bound approach as a bounding function. The ACO-DBM outlined in this chapter builds upon the concept of an idealized laminate. Whilst the MPSO does not directly contain a local search enhancement, the improvements outlined in the next section give greater scope to the search process. Finally, the SDGD, as it builds upon the DBM, is indeed a local search approach. In the following analysis, measuring the improvements in efficiency, reliability and robustness are key. Next, the first of the three new approaches to determine lay-ups is introduced. Namely, a modified particle swarm optimization.

5.3 Modified Particle Swarm Optimization

As discussed in Chapter 4, the goal of the second level of the optimization is to determine a laminate stacking sequence which satisfies the set of design constraints. This is achieved using a modified particle swarm optimizer (MPSO) as a discrete optimizer. The MPSO builds upon the standard PSO introduced in Chapter 4. A standard particle swarm algorithm contains a swarm of particles (lay-ups), where each particle represents a potential solution to the optimization problem. Particles move through a multidimensional search space where the position of the particle is adjusted according to its own experience and also the experience of the other particles in the swarm. To remind the reader, details concerning the PSO are outlined here. Let $x_{ij}(t)$ denote the jth ply (from the outer surface to mid-surface) in the ith lay-up at the tth iteration. The position of the particle (lay-up) is updated by adding a velocity vector, $v_{ij}(t)$ to the current position, i.e.

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1)\Delta t \tag{5.1}$$

where,

$$v_{ij}(t+1) = \omega \cdot v_{ij}(t) + \frac{c_1 r_1(t) \cdot (L_{ij}(t) - x_{ij}(t)) + c_2 r_2(t) \cdot (G_j(t) - x_{ij}(t))}{\Delta t} \tag{5.2}$$

where ω, c_i are user-defined constants used to accelerate the swarm and control the 'inertia' of the swarm, respectively and $\Delta t = 1$. Functions $r_1(t), r_2(t)$ are random numbers generated within the interval [0, 1], $L_{ij}(t)$ is the position (local best) of the jth ply in the i th lay-up in its history over all iterations:

$$L_{ij}(t)=\begin{cases} L_{ij}(t-1) & \text{if} \quad F(x_{ij}(t))\geq F(L_{ij}(t-1)) \\ x_{ij}(t) & \quad\quad F(x_{ij}(t))<F(L_{ij}(t-1) \end{cases} \tag{5.3}$$

where F is the fitness function which is to be minimized (see Chapter 3) and is,

$$F(x_i)=\max_i\{G_i\} \tag{5.4}$$

where i is the ith constraint. The global best lay-up, denoted by G is calculated as,

$$G(t)=\arg\min\{F(L_i(t))\} \;\; \forall i,t \tag{5.5}$$

By updating the position vector with the velocity vector over a small time period (iteration) using Eqn. (5.2), each lay-up is either constrained to stay within its own neighbourhood or is 'forced' to a different area of the search space. Note, the velocity vector $v_{ij}(t)$ drives the optimization and thus the convergence of the swarm to an optimal solution. The velocity vector $v_{ij}(t)$ also represents the change in ply angle of the jth ply in the ith lay-up in dimension. In the standard particle swarm outlined above, there is a tendency for the PSO to converge to points, which may or may not be local optima. This is evident where the fitness function is multi-modal and the response surface is non-smooth. For a continuous and differentiable function, f, a gradient based optimizer can be used to obtain a local minima. To utilize both a gradient and swarm, Thompson et al. (2001) proposed a pseudo gradient approach. Let,

$$x'_{ij}(t+1)=x_{ij}(t)+v_{ij}(t+1) \tag{5.6}$$

$$x''_{ij}(t+1)=x_{ij}(t)+\beta\cdot v_{ij}(t+1) \tag{5.7}$$

then,

$$x_{ij}(t+1)=\arg\max\left(\frac{-f(x'_{ij}(t+1))-f(x_{ij}(t))}{\beta},\frac{-f(x''_{ij}(t+1))-f(x_{ij}(t))}{\beta}\right) \tag{5.8}$$

where $\beta \in (0,1)$ and is randomly generated at each iteration. Engelbrecht (2003) observed that whilst this method was suitable for a unimodal problem, it may suffer for a complex multi-modal problem. Note, the problems considered in this thesis are indeed multi-modal (e.g. non-convex) when ply orientations are used as design variables. Motivated by this shortfall, a new velocity function is proposed,

$$v_{ij}(t+1) = \eta \cdot (G_{1j}(t) - x_{ij}(t)) \tag{5.9}$$

where η is a pseudo-random scalar drawn from a normal distribution with mean 0 and standard deviation 1. Note, η gives greater diversity to the population since it is not simply bounded between [0,1]. It can be seen from Eqn. (5.9) that the swarm is stochastically attracted to the current global best. That is, the swarm will move in the direction of the global best, but the utilization of η may yield new designs by varying the population. Furthermore, to prevent premature convergence, if the fitness of the current global best position approximates the mean fitness of the swarm the velocities are re-initialized at random positions. The velocity of the swarm is also re-initialized if the global best has not changed over ten iterations. It is observed that the resulting MPSO is parameter free. That is, the user is not required to define, a priori, the standard PSO parameters c_1 and c_2 shown in Eqn. (5.2). The effectiveness of the MPSO over the standard PSO is highlighted in section 5.5 which contains numerical examples. However, before this, it is necessary to introduce the second key development to the second level optimization process.

5.4 Combined Ant Colony Direct Branching Method

As previously mentioned, the goal of second level of the optimization is to determine a stacking sequence which satisfies the set of constraints. Depending upon the number of constraints, several methods are employed. If the number of constraints equals one, it is now shown that a Greedy Search, called a Direct Branch Method (DBM) herein, using a so-called idealized laminate maximizes the constraint reserve factor and thus obtains the optimum lay-up. This is later proved. In contrast, if the number of constraints is greater than one, a combined ACO-DBM approach can be utilized. The presentation, discussion and evaluation of this new method are the focus of this section.

5.4.1 Direct Branching Method

The purpose of the DBM is to improve the fitness of a given stacking sequence using a shortest path approach. In order to use the DBM, where there exists just one constraint, a laminate idealization is introduced. It is observed that Kim and Hwang (2003) defined an ideal lamina to be one with enhanced stiffnesses that are unachievable, yet relate to the inherent basic ply stiffnesses. It is noted that Kim and Hwang (2003) restricted their study of ideal laminae to symmetric orthotropic/isotropic laminates. However, a new definition of the ideal lamina is made which makes use of the optimum lamination parameters and thicknesses obtained at the first level of the optimization and thus extend the concept of the idealization to anisotropic laminates. First the reduced stiffness for the *i*th layer is defined as,

$$
\begin{aligned}
Q_A^i &= A / h \\
Q_B^i &= 4B / h^2 \\
Q_D^i &= 12D / h^3
\end{aligned}
\qquad \textbf{(5.10)}
$$

where the A, B, D matrices are calculated using Eqns (2.2-2.4) with the optimum lamination parameters determined from the first level optimization. and h is the rounded (up) plate thickness obtained from the first level of the optimization (detailed in Chapter 3). By selecting the above idealization, the DBM initially begins by assuming the laminate has the aforementioned optimum properties. Each ideal layer in the laminate has the value defined in Eqns. (5.10). Next, the DBM algorithm is detailed and the scheme is shown in Fig. 5.1, where i represents an idealized layer.

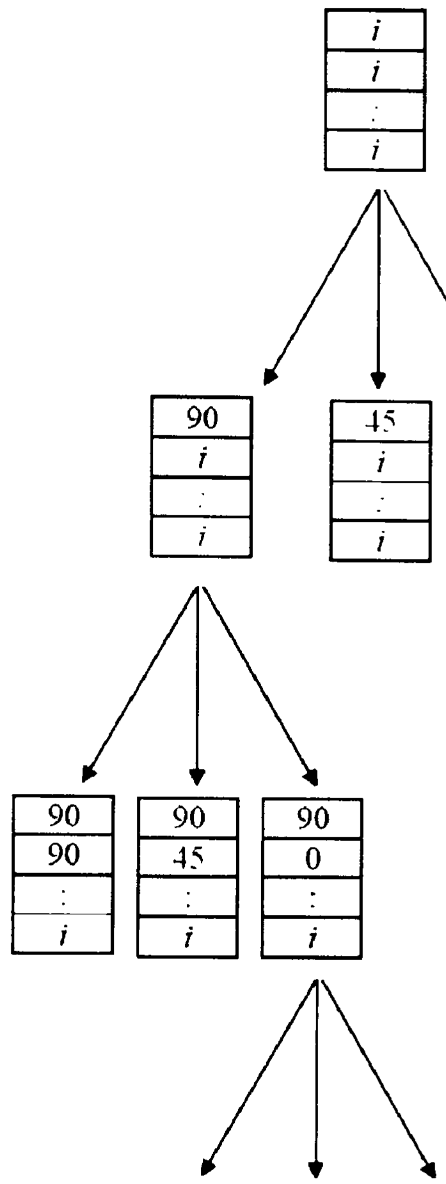

Fig. 5.1 - Scheme of DBM with 3 possible ply orientations [90/45/0]

Note, the DBM stimulates a greedy algorithm and is summarised in Algorithm 5.1,

Algorithm 5.1

1) Begin with an initial idealized lay-up (if there is one constraint, use an idealized laminate, else use a randomly generated lay-up)

(2) Starting from the outermost layer, substitute the first ply with every other possible ply orientation in the design set and evaluate the fitness function with each newly obtained lay-up.

3) Select the angle corresponding to the best fitness and then proceed to the next ply.

At any step in the algorithm (where a step in the path corresponds to one ply) where the constraint(s) are satisfied the algorithm is immediately terminated and the stacking sequence returned. Note, at this level the minimization of a particular constraint is not sought, simply that the constraints are satisfied. Furthermore, the DBM will only terminate at the end if an idealization is used. This is because until the final ply has been selected, the laminate remains partially idealized. Interestingly, the DBM has two particular features. For a single constraint, using the idealization shown above, the DBM maximizes the constraint reserve factor. Furthermore, if the starting point in the DBM is a real stacking sequence and not an idealized one, then it is easily proved that for a given stacking sequence the DBM will either improve the fitness of the stacking sequence or at worse, not change it. Steps 2 and 3 demonstrate this feature. If the number of constraints is greater than one, there may exist complex trade-offs between the constraints. As such, the above method may not be sufficient to determine lay-ups which satisfy the constraints. Motivated by this deficiency a combined ACO-DBM is adopted. In Fig. 5.2, the fitness of the laminate is shown by varying the outermost ply for the buckling (blue curve) and strength constraint (red curve).

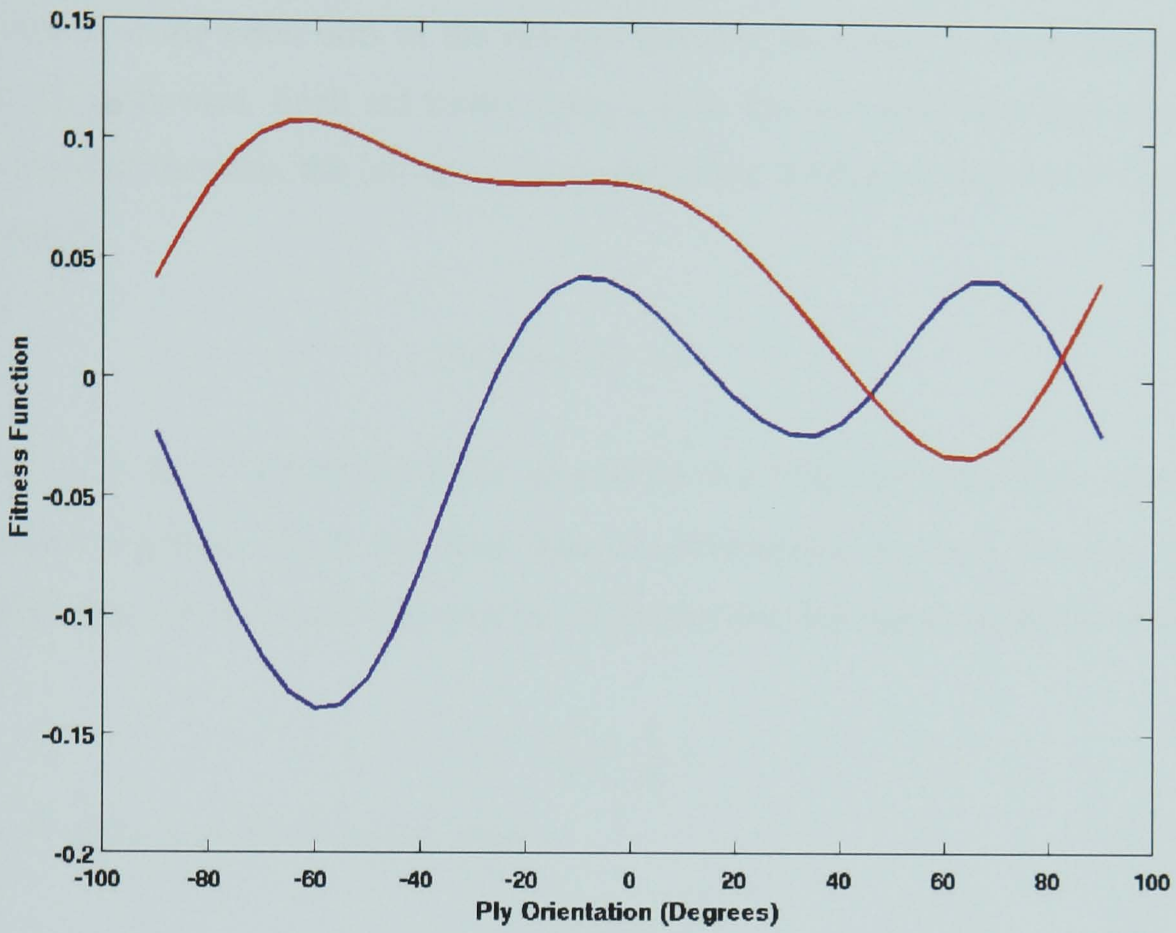

Fig. 5.2 - Laminate fitness generated by varying the ply orientation of the outermost layer assuming the remaining layers are ideal

In Fig. 5.2, it is shown that using ply orientations as the design variables yields a complex non-convex response surface. Additionally, the difference in the fitness function at each ply orientation (for the buckling and strength constraint) highlights the complex trade-offs at ply level between each constraint. Therefore, in multi-constraint optimization, no guarantee can be made that DBM will determine a design which solves the CSP. However, if one constraint is used, the DBM will maximises the constraint reserve factor. Next, the Ant Colony meta-heuristic is re-introduced for current purposes.

5.4.2 Ant Colony Optimization

To remind the reader, the ACO meta-heuristic is outlined again. An ACO (Dorigo 1993) contains a colony of ants, where each ant represents a notional traveller that traverses each step in a path. This is analogous to an ant walking and traversing each ply in a lay-up. Ants move through a multidimensional search space where the dimension equals the number of plies (half the number of plies for a symmetric lay-up). At each iteration, an ant selects a path (lay-up) which changes according to its own experience and also the experience of the other ants in the colony. Initially, the paths of the k ants are pseudo-randomly generated. Each ant walks along a path (lay-up) such that there are k lay-ups. At the end of the path, the lay-up, is evaluated using the fitness function F (introduced in Chapter 3),

$$F(x) = \max_i \{G_i(x)\} \tag{5.11}$$

and $G_i(x)$ is the i^{th} constraint at the second level noting that a positive value denotes a constraint violation. Each ant then deposits pheromone on each arc (ply in a given position) that it has visited. The amount of pheromone deposited is calculated as follows,

$$\tau_{ij}^k = \frac{1}{F_k} \tag{5.12}$$

if the i^{th} angle is in the k^{th} lay-up, else

$$\tau_{ij}^k = 0 \tag{5.13}$$

where τ_{ij}^k is the pheromone matrix for the k^{th} ant, $i = 1 .. \Phi$, where Φ is the number of angles in the design envelope and $j = 1 ... r$ where r (half the number of plies. Observe, the definition of the pheromone matrix given in Eqns. (5.12-5.13) ensures that a low fitness equates to a higher pheromone deposit and vice versa. Note, τ_{ij}^k cannot be infinite since the stopping condition is applied before evaluation. Note, if the value of the fitness function is zero, the algorithm is terminated. The next step in the algorithm is the calculation of the ant routing table, ψ_{ij}. Let,

$$a_{ij} = \sum_{l=1}^{k} \tau_{ij} \tag{5.14}$$

then,

$$\psi_{ij} = a_{ij} \quad if \quad a_{ij} > 0 \tag{5.15}$$

else

$$\psi_{ij} = \max_i (a_i) \tag{5.16}$$

When an ant begins its path, it decides which ply orientation to select using the information contained in Eqns. (5.14-5.16) .Each ant then walks along a new path (lay-up_. At each ply, the ply orientation is selected using the following probability function,

$$p_{ij} = \frac{\psi_{ij}}{\sum_{i=1}^{\Phi} \psi_{ij}} \tag{5.17}$$

Each ply angle i at position j in each lay-up is randomly selected with a probability p_{ij} which is calculated using Eqn. (5.17). In practice, the process of selecting ply orientations with given probabilities is achieved in MATLAB (2009) using the 'randsrc' function. It is noted that the process of selecting ply orientations is stochastic and thus having a higher probability does not guarantee selection. Furthermore, at each iteration, the DBM is applied to the best ant to seek local refinement. The block structure of the combined ACO-DBM is showed in Fig 5.3.

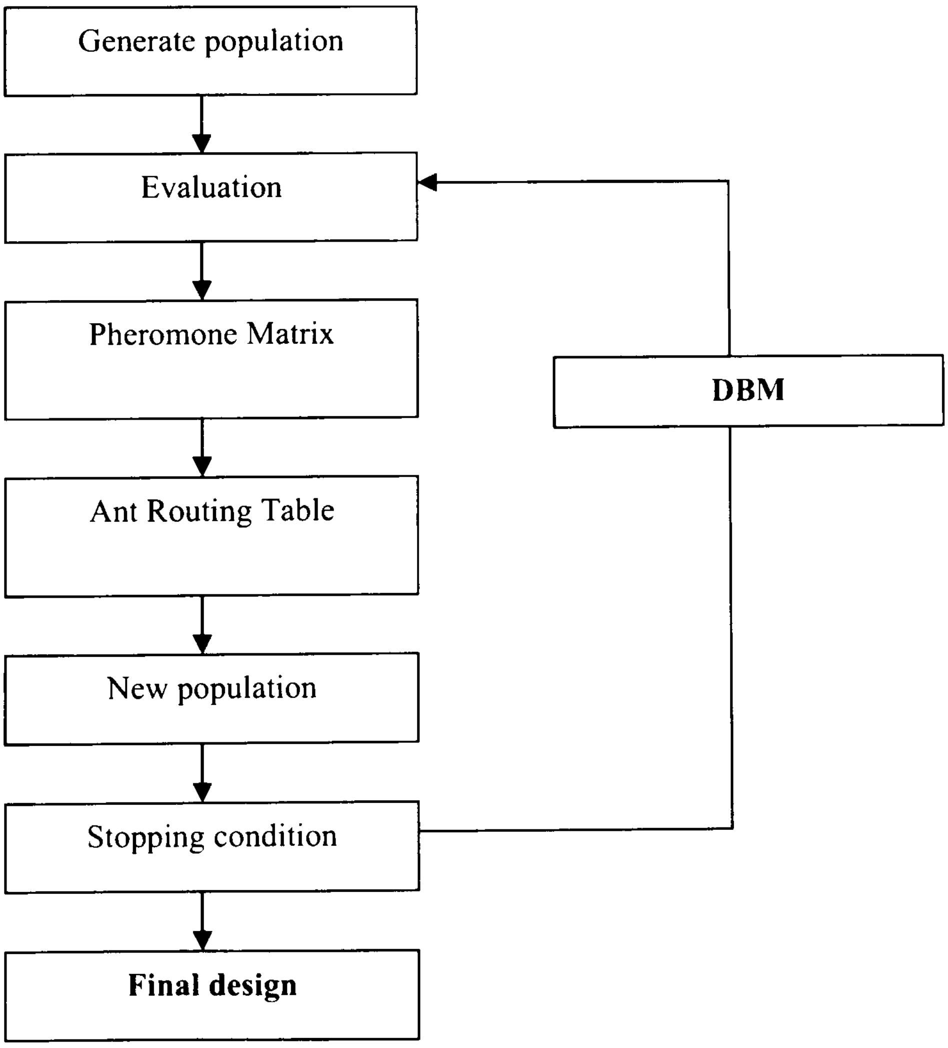

Fig. 5.3 - ACO-DBM Block Structure

At the next iteration, pheromone concentration is then updated (for good designs) using the following function,

$$\tau_{ij}^{k}(t) \leftarrow \tau_{ij}^{k}(t)+\tau_{ij}^{k}(t-1) \quad \text{if} \quad F_k(t) < F_k(t-1) \tag{5.18}$$

else,

$$\tau_{ij}^{k}(t) = \tau_{ij}^{k}(t-1) \tag{5.19}$$

where t is the t^{th} iteration. It is noted that the pheromone matrix is only updated when the solution in the t^{th} iteration is superior to the $t-1$ solution. Note, in the above ACO no heuristic information is used. As such, the above ACO represents a more simplified version of the standard ACO detailed by Aymerich and Serra (2008).

In the ACO-DBM implemented in this thesis the selected population size is 20. However, in general, let the population number be r and the number of iterations be s. It follows that the total number of evaluations in the ACO is r x s. At each iteration, the DBM is called, m x n stacking sequences are evaluated, where there are m different ply orientations and n plies in the stacking sequence. This is equivalent to (m x n)/r iterations when running the ACO in isolation. As such, the total number of evaluations in the combined ACO-DBM is r x (s+mn). Note, the second level is highly efficient when closed formed solutions are used. If closed form solutions are not available, first or second order approximations of the constraints can be formed using the optimum design variables and associated sensitivities obtained at the end of the first level. Next, it is necessary to prove that the DBM obtains local optima.

Theorem

Let f denote a single constraint and let f_k^{ID} denote the fitness of the ideal laminate (stacking sequence) with k real layers determined, then it follows that by the using the DBM, local optima are obtained.

Proof

First, it is proven below that the DBM maximizes the constraint reserve factor for a single constraint. Let f denote a single constraint and let f_k^{ID} denote the fitness of the ideal laminate (stacking sequence) with k real layers determined where $0 \leq k \leq n$ and n is the total number of plies in the laminate. Next, an operator '$\circ$' is defined which denotes 'the point geometrically closest to'. Hence, the expression $a \circ b$ implies that a is the closest point geometrically to b. Defining the first ply in the laminate,

$$\theta_1 \rightarrow \min\left|f_1^{ID} - f_0^{ID}\right| \;\forall\; \theta \in \varphi \tag{5.20}$$

noting φ is the set of possible ply orientations and f is a scalar function. In general,

$$\theta_i \rightarrow \min\left|f_i^{ID} - f_{i-1}^{ID}\right| \;\forall\; \theta \in \varphi \tag{5.21}$$

Once the first ply is selected, it is clear that $f_1^{ID} \circ f_0^{ID}$. Once the second ply is determined,

$$f_2^{ID} \circ f_1^{ID} \circ f_0^{ID} \tag{5.22}$$

In general,

$$f_n^{ID} \circ f_{n-1}^{ID} \circ \ldots \circ f_1^{ID} \circ f_0^{ID} \tag{5.23}$$

It follows,

$$f_n^{ID} \circ f_0^{ID} \tag{5.24}$$

∎

Note f_n^{ID} corresponds to the fitness of a real stacking sequence since all n ideal layers have been substituted by real layers. Additionally, the DBM can be viewed as an optimality criteria. For each ply in the lay-up (assuming the other plies remain at their current value), there exists no ply in the set of possible ply orientations which will improve the value of the fitness function. This process is analogous to a gradient optimizer which perturbates about the current point by a fixed value. For example, if the gradient of the Lagrangian is zero (see Chapter 3), then it is assumed that the vector of design variables is an optimum configuration. However, with the DBM it is impossible to take the gradient of the objective function as a) the variables take only discrete values (in the examples in this thesis) and b) the fitness function (fitness function outlined in Chapter 3) is discontinuous.

In summary, a novel ACO-DBM approach has been presented to solve the second level optimization problem. The method builds upon the standard ACO which was shown to be excellent in locating the vicinity of good lay-ups, but sometimes poor at local refinement. By introducing a DBM, local refinement (or search) can easily be exploited. As such, a novel combined ACO-DBM is proposed. Additionally, where there exists only one constraint in the second level optimization, through the easy introduction of a so called ideal laminate, the lay-up which maximises the constraint reserve factor is readily obtained. The introduction of the DBM resulted in several interesting observations. In particular, how the DBM can be viewed as an optimality criteria. Motivated by this and the stochastic nature of meta-heuristics, a stochastic gradient descent optimizer is presented in the next section. Following this, the set of numerical examples are given.

5.5 Stochastic Discrete Gradient Descent

In section 5.4 it was proven that the DBM would find the lay-up which maximises a single constraint reserve factor assuming a laminate idealization is used. However, when the number of constraints was greater than one, there was a high likelihood that a DBM alone would be incapable of finding a solution. This is partly due to the complex trade-offs between constraints at a ply level. Whilst a branch and bound approach may be induced at the point when feasibility is broken, it is generally observed that a branch and bound maybe more inefficient that documented meta-heuristic techniques. Using the DBM, but with the inclusion of a stochastic element, a new approach which simulates the behaviour of a gradient based optimizer is now presented. The core steps of the process, named an SDGD, are outlined in Algorithm 5.2.

Algorithm 5.2

1) For each ply (from the outermost ply inwards), evaluate the angle before and after the current ply orientation in the enumerated set of ply orientations. Note if the existing ply orientation is the min or max of the set, then only one angle is considered.
2) Determine the orientation which minimises the fitness function for the given layer and change the corresponding ply position in the original lay-up,

$$\theta = \arg\min(F) \tag{5.25}$$

3) The new lay-up becomes the next starting point of the SDGD
4) Repeat steps 1-3.
5) If there is no change in the fitness function and the CSP is not solved, randomly change a ply orientation.
6) Repeat from step 1 until the minimum reserve factor is greater or equal to one.

Note, at any step in the process, if the minimum reserve factor is greater or equal to one, the algorithm is terminated. To remind the reader, the algorithm does not seek, directly, to minimise a given function. Rather, the algorithm seeks to determine a lay-up which satisfies the aforementioned CSP. Note, if one full iteration completes without improving the lay-up, it is deemed to have found a local optimum which does not satisfy the CSP. To escape local optima (and without using a population based heuristic), a randomly selected ply angle is randomly altered to introduce a stochastic element to the process. Next, the SDGS is initiated again with the new stacking sequence acting as the starting point. If the objective function does not change over a threshold number of iterations (30) then the optimization is terminated to avoid unnecessary computational time. Algorithm 5.2 is now demonstrated through an example. In the following example, the enumerated (lowest value to highest) set of ply orientations is $-60, -45, -30, 0, 30, 45, 60, 90$. The lay-up is symmetrically laminated. Suppose the starting point is a randomly generated laminate:

$$S_1 = [60/45/-45/30/0/90/90]_S \tag{5.26}$$

The first key step of the SDGD is to vary the first ply (from the outermost ply inwards). The lower bound nearest neighbour (LBNN) is defined as the ply below the current ply in the enumerated set. Therefore,

$$S_2 = [45/45/-45/30/0/90/90]_S \quad \textbf{(5.27)}$$

The upper bound nearest neighbour (UBNN) is analogously calculated,

$$S_3 = [90/45/-45/30/0/90/90]_S \quad \textbf{(5.28)}$$

Note, if the following condition holds,

$$f(S_2) \leq f(S_1) \leq f(S_3) \quad \textbf{(5.29)}$$

then the ply position in question is said to have a discrete gradient of zero. This is because the value of the objective function is bounded above and below. This is analogous to the gradient of a non-linear convex function which is bounded above and below by its directional derivatives. Next, the second ply is changed. The LBNN is,

$$S_4 = [60/0/-45/30/0/90/90]_S \quad \textbf{(5.30)}$$

Conversely, the UBNN,

$$S_5 = [60/60/-45/30/0/90/90]_S \quad \textbf{(5.31)}$$

Note, the final two stacking sequences which are considered in the first iteration are

$$S_{n-1} = [60/45/-45/30/0/90/60]_S \quad \textbf{(5.32)}$$

and

$$S_n = [60/45/-45/30/0/90/90]_S \quad \textbf{(5.33)}$$

From the n stacking sequences evaluated, the one which minimises the objective function is chosen. Consequently, a single ply in S_1 is changed. Once the first iteration is complete, the process repeats and the stacking sequence, which has a correspondingly lower objective function, becomes the new starting point. In the SDGD, if the current ply position is at the min or max of the enumerated set, then only the UBNN and LBNN are computed, respectively. When Eqn. (5.30) holds for each ply in the lay-up, then the stacking sequence is said to be a local optimum. If the lay-up does not solve the CSP, then a stochastic element is introduced. That is, a randomly selected ply (using the *randsrc* function in MATLAB) is changed. This enables the SDGD to converge, albeit stochastically, towards a feasible solution.

In Chapter 3, a gradient based optimization was introduced. In the detailed approach, the gradient of each variable with respect to the Lagrangian is computed. If a finite difference approximation (of the gradients) is used, then this represents $n+1$ function evaluations where n is the number of design variables. In contrast, if central differences are used, this results in $2n+1$ evaluation per iteration. As such, the SDGD mimics the central difference approach (only if a numerical optimization is used). Consequently, the method is computationally equivalent (as defined in Chapter 4) to a gradient based optimizer outlined in previous chapters. Note, the approach is not a population based technique and ultimately depends upon a stochastic operator (generating a random ply) to escape local optima. Whilst the SGD offers a simplistic approach to lay-up optimization, the number of function evaluations required to determine a suitable solution may be comparable to a standard ACO or PSO outlined in Chapter 4. In the next section, a number of numerical examples relating to the MPSO, ACO-DBM and SDGD are presented.

5.6 Numerical Examples

In this section, a number of numerical examples are provided. The motivation is to demonstrate the significant efficiency and functionality gains associated with the introduction of the MPSO and ACO-DBM. Additionally, the numerical examples seek to demonstrate the effectiveness of the SDGD in obtaining feasible solutions to the CSP. The following examples are taken from Chapter 4 to serve as a benchmark. To remind the reader, three different sets of ply orientations are considered,

1) $[0, 90, \pm 45, \pm 30, \pm 60]$ degrees
2) $[-90, 90]$ in 15 degree increments
3) $[-90, 90]$ in 5 degree increments

For each case study, a single load case is selected– this being a simply supported rectangular composite plate of dimensions 1200mm x 200mm. The selected load case is:

1) $Nx = 0, Nxy = -700$ N/mm

Similar to Chapter 4, the ply thickness is artificially changed to alter the thickness of the laminate In the following numerical examples, the following attributes are considered: reliability (total number of successes in finding a solution divided by total number of runs), robustness (range of obtained reserve factors) and efficiency (average number of iterations taken to determine a solution). Note, for each sub-problem the optimization algorithms were run 100 times to obtained averaged results. This was deemed sufficient in order to assess the degree of randomness in each meta-heuristic technique. The maximum number of iterations for each method was 200. The stopping criterion is if the laminate had reserve factors of greater or equal to one for the structural constraints and thus satisfying the CSP. Initially, results from the MPSO are considered

5.6.1 Modified Particle Swarm Optimization

In Table 5.1, the PSO and MPSO are compared. Note, the MPSO contains a population size of 20 and a stochastic parameter η.

Table 5.1 – Comparison between PSO and MPSO

Number of Plies (T=2*NP)	*Ply Orientation Set*	*Efficiency*		*Reliability*		*Robustness*		*Function Evaluations*	
		PSO	*MPSO*	*PSO*	*MPSO*	*PSO*	*MPSO*	*PSO*	*MPSO*
8	1	23	20	100	100	1	1.02	310	291
16	1	20	18	98	100	1	1.1	610	490
32	1	29	22	98	100	1	1.01	1793	1605
10	2	14	13	100	100	1	1	200	190
21	2	17	15	99	100	1.01	1.04	610	493
42	2	28	21	100	100	1	1.1	1501	1021
10	3	35	32	99	100	1.01	1.03	391	290
20	3	14	11	99	100	1	1	987	771
30	3	29	17	100	100	1	1	1602	890

From Table 5.1, it is clear that the MPSO leads to significant efficiency and functionality savings. The gains were achieved for all sets of ply orientations. However, the efficiency savings were considerable when the set of ply orientations approximated a near continuous set of ply orientations. This supports the analysis in Chapter 4 which stated that whilst the PSO was an improvement over the standard GA, it was clear that PSO was better suited for near continuous problems. The same applies to the MPSO. Further analysis of these results is given in section 5.7. Next, results from the ACO-DBM are considered.

5.6.2 ACO-DBM

In Table 5.2, a comparison is made between the ACO and the ACO-DBM. As the DBM introduces additional computational expense (additional number of stacking sequences are evaluated at each iteration), attention is paid to the number of function evolutions`. Note the ACO parameter values defined in Chapter 4 are applied to the ACO-DBM detailed in this Chapter.

Table 5.2 - Comparison between ACO and ACO-DBM

Number of Plies	*Ply Orientation Set*	*Efficiency*		*Reliability*		*Robustness*		*Function Evaluations*	
		ACO	*ACO-DBM*	*ACO*	*ACO-DBM*	*ACO*	*ACO-DBM*	*ACO*	*ACO-DBM*
8	1	19	14	99	100	1.01	1.02	520	400
16	1	23	12	98	100	1	1.04	680	500
32	1	27	15	98	100	1	1.01	1487	1010
10	2	11	9	100	100	1.02	1.03	189	170
21	2	15	6	100	100	1.01	1.01	403	371
42	2	23	11	98	100	1.01	1	1331	1007
10	3	31	17	99	100	1.01	1	1541	1395
20	3	14	10	99	100	1	1	450	371
30	3	12	10	100	100	1	1	389	345

Upon inspection of the above results, it is clear that the ACO-DBM leads to increased efficiency and functionality. The implications on the number of function evaluations will be discussed in section 5.7. Next, results from the SDGD are presented.

5.6.3 SDGM

The results for the SDGD are given in Table 5.3. Note, as no standard version existed no direct comparison is made in the same way the PSO is compared with the MPSO etc. Furthermore, the SDGD contains no parameters including population size. This is because the SDGD utilizes only a single stacking sequence.

Table 5.3 – Results obtained from SDGD

Number of Plies	*Ply Orientation Set*	*Efficiency*	*Reliability*	*Robustness*	*Function Evaluations*
		SDGD	*SDGD*	*SDGD*	*SDGD*
8	1	20	100	1.02	340
16	1	19	99	1	627
32	1	29	100	1.01	1885
10	2	11	100	1.03	231
21	2	12	98	1.01	516
42	2	18	100	1.05	1530
10	3	19	99	1.01	399
20	3	29	100	1	1189
30	3	31	100	1.02	1891

The results from the SDGD imply that the method was robust and reliable. Whilst the SDGD is generally less efficient than the ACO-DBM or MPSO, it was successful in obtaining good solutions. Further observations are discussed in the next section. Following the discussions of results, conclusions are drawn.

5. 7 Discussions of Results

From Table 5.1, the MPSO outperformed the PSO across the board. On average the number of function evaluations decreased by over 15 percent. Furthermore, the reliability of the MPSO was close to 100%. Additionally, the reliability shows that the MPSO was able to determine good designs which solve the CSP. When comparing the MPSO and standard PSO, the obtained results are readily understood. It is observed that Eqn. (5.9) allows the rapid improvement of the swarm by giving greater diversity to the population (amount of change). Moreover, it enables the swarm to converge quickly to stable points. Additionally, by using the pseudo-gradient proposed by Thompson et.al (2003) the swarm, stochastically, moves towards global minima – by mimicking a gradient optimizer but utilizing swarm behaviour to search a wider domain. It is thought that the combination of the new velocity function, gradients and swarm re-initialization significantly builds upon the standard PSO in determining laminate stacking sequences. As previously mentioned, the lay-ups which are obtained when running the three aforementioned optimization algorithms do not necessarily satisfy common industrial constraints such as the 4-ply contiguity rule. However, they can be readily included, as detailed in Herencia et al. (2008). In general, the MPSO (plus ACO-DBM and SDGD) allows for the inclusion of additional constraints and different sets of ply orientations both which may ultimately affect the final design and the performance of the algorithm. In Chapter 7, it is recommended that an area for future work may be the benchmarking of the new algorithms with the inclusion of common industrial design rules.

With respect to the ACO-DBM, from Table 5.2, it can be seen that the inclusion of the DBM with ACO significantly reduced the overall number of iterations to determine a feasible design highlighting the potential efficiency savings obtained with the aforementioned framework. The number of function evaluations is slightly less than the standard ACO. However, the quality of design obtained by the ACO-DBM is on average 10% better than the standard ACO with average efficiency savings of circa 40%. In summary, the DBM acted, as designed, to give a local search once the ACO had determined a decent solution. In the current version of the ACO-DBM, the DBM is applied at each iteration. In contrast, it may be more appropriate to apply the DBM once the ACO is deemed to have converged, e.g. little or no change over the fitness function for a number of iterations. Again, this may be the focus of future work.

As previously stated, the derivation of SDGD was motivated by analysis of the DBM. The SDGD was successful at determining feasible quality solutions to the CSP demonstrated by Table 5.3. Whilst the number of function evaluations was comparable to both the ACO-DBM and PSO, the number of iterations was generally larger than the other methods. However, the SDGD was shown to be robust and reliable. Note, that if only one constraint exists, the DBM will determine a solution in m x n evaluations. Next, some conclusions are made based upon the analysis and results obtained in this chapter.

5.8 Conclusions

In this chapter an MPSO and ACO-DBM have been presented. Motivated by the analysis of the DBM a stochastic gradient descent method was introduced. Both the MPSO and ACO-DBM were shown to be significant improvements from their respective original variants for this particular purpose which is the identification of a stacking sequence to satisfy the set of structural constraints. Additionally, the SDGD was shown to be effective at determining solutions to the CSP but was generally less efficient than the ACO-DBM and MPSO as the set of ply orientations tend to a continuous set. In summary, when the set of ply orientations is inherently discrete, e.g. the incremental change between possible ply orientations is greater than 5 degrees, a two-level approach integrating the gradient based optimization and the ACO-DBM is advocated for the aforementioned reasons. In contrast, if the change in possible ply orientations is less than 5 degrees and two-level approach with a gradient optimizer (at the first level) then an MPSO at the second level would be more suitable. Furthermore, if only one constraint exists, then DBM can be used with a laminate idealization.

In the next Chapter, the optimization approach outlined in this thesis is expanded for multi-part laminated composite structures. A conceptual approach is presented for the parallel optimization of multi-part laminated composite structures.

Chapter 6
Parallel Optimization of Multi-Part Structures

6.1 Introduction

In Chapters 3-5, a two-level optimization for laminated composite plates was presented. At the first level, lamination parameters and plate thicknesses were used as design variables to minimize the mass of the plate subject to local structural constraints. To achieve this, a gradient based method was used. At the second level, a discrete optimizer was used to identify a stacking sequence which satisfies the set structural constraints. Despite the demonstrated success of this approach, the examples were limited to single composite plates. The focus of the current chapter is to expand this approach, utilising parallel computing, to derive a conceptual and computationally efficient approach to optimize laminated multi-part composite structures.

Currently, there are practically no computationally efficient optimization tools that provide detailed structural analysis for multi-part laminated composite structures. This shortcoming is due partly to the explosion in number of design variables with an increasing number of parts leading to increased computational complexity. To overcome this potential problem, a two-level parallel optimization approach of laminated composite structures is proposed. The approach builds upon the work outlined in previous chapters. At the first level, the structure, an idealised composite wingbox is decomposed into plate elements (plates). Local design variables are then assigned to each plate, namely, lamination parameters and local thicknesses. These design variables are then used to minimize the mass of the wingbox using a gradient based method as outlined in Chapter 3. Gradient calculations/sensitivities are performed in parallel using a number of processors. At the second level of the parallel optimization, a combined ACO-DBM, detailed in Chapter 5, is used to determine laminate stacking sequences. In parallel, a lay-up is determined which satisfies the local design constraints. The internal loads in the wingbox are then recalculated. If the change in internal loads between the current and optimal configuration is less than a specified tolerance, the final design is said to be determined. Otherwise, the ACO-DBM is relaunched to find a lay-up which satisfy the constraints (and the new internal loads).

The proposed method, whilst conceptual, offers a potentially reliable and efficient approach to the optimization of multi-part laminated composite structures partly due to its scalability. Note, the purpose of this chapter is not to undertake a detailed analysis of a multi-part laminated composite structure. Rather, the aim is to provide a broad framework and foundation for a two-level optimization using lamination parameters. Furthermore, the aim is to provide a conceptual framework building upon the primary objectives of this thesis rather than provide detailed numerical examples. In sum, the approach outlined herein may reduce the running time for the optimization and produces an effective way of undertaking multi-part laminated composite structural optimization.

6.2 Literature Review

The use of parallel computing to solve computationally intensive tasks has been popular in recent years. Due to the sheer number of variables in multi-part laminated composite design, parallelization offers an attractive route to improve the efficiency of the process without sacrificing the quality of the optimization and ultimately the design. The current trend in research within both academia and industry is the use of high performance parallel computation (HPC). In composite design optimization, the use of parallel computation is seen as effective due to the computational cost associated with finite element analysis (FEA). It is generally accepted that the computational cost associated with FEA is one of the greatest hindrances to structural optimization. This computation time significantly increases as the complexity or size of the structure grows. Whilst close formed solutions offer invaluable insight and help to reduce the impact of FEA processing, such solutions are not always readily available for complex structural interaction.

It is observed that HPC can be applied to the design and optimization of composite laminates at two separate levels. Firstly, with respect to FEA, parallel computation can be used to run an analysis in a quicker amount of time without loss of accuracy. This allows for the analysis of more complex structures such as wing boxes or entire wing structures. At the FEA level, FEA programs such as MSC NASTRAN (Patel 2002) incorporate the superelements function. Using superelements, the initial size of the problem is reduced into smaller segments, which can be computed independently. This independence can be used with HPC to create an almost 'ideal situation'. The independence allows the decoupling of a large structural problem into small sub problems. These sub problems require significantly less computation time and often avoid potential 'crashes' due to the exceeding of memory allocation. HPC allows each of the sub problems to be run on a separate processor. In sum, this approach reduces the running time of the individual analyses without a compromise in accuracy. Motivated by this, the approach proposed in the Chapter utilises a decomposition approach yet the superelements function is not used since the structural examples are simplistic in nature and furthermore, local design constraints can be formulated in terms of CFS rather than potentially expensive finite element computations.

As noted, parallel computation can be used to reduce the processing time for FEA of complex or large structures. Furthermore, parallel processing can also be applied to direct and heuristic search techniques. Henderson et al. (2004) studied the use of a parallel genetic algorithm for stacking sequence optimization of a composite laminate subject to buckling, strength (allowable strain) and ply contiguity constraints. Briefly, the authors used parallel genetic algorithms were used to search the design space. The searches took place independently. Once the each GA had converged, each optima from each GA was assessed accordingly. The global optima was determined to be minimum value of all local optima determined by running the parallel GA. Note, this approach does not guarantee that the global optima is found only that the probability of finding the global optima is increased. This method was slightly different to that of Punch et al (2006) who proposed the design and optimization of composite structures using coarse-grain parallel genetic algorithms. The authors implemented a new coarse-grain parallel architecture for genetic algorithms, called island injection genetic algorithms (where data is communicated infrequently usually after significant amounts of computation). This approach fine-tunes each generation of the population in the GA. The authors showed that by using a parallel GA with an injection algorithm, a speed up in the optimization was achieved with similar results in quality. Furthermore, Gurdal et al. (2004) used a parallel GA with migration to optimize a delta wing structure. The migratory GA allowed for the effective use of ply continuity constraints. Whilst the use of parallel optimization reduced the overall run time, the method was still inefficient due to the number of evaluations required in the GA for convergence. This was partly due to ply continuity constraints. More recently, Setoohdeh et al. (2007) used lamination parameters in conjunction with parallel computation and the cellular automata (CA) heuristic to optimize a variable stiffness plate for in-plane response. The authors demonstrated significant efficiency gains in using a parallel approach. They did not restrict their work to finite sets of angles (in contrast to this thesis), but rather the complete feasible region of lamination parameters was used. As discussed in Chapter 2, this region is in-fact a large number of linear approximations to the feasible region. A large number of constraints will naturally reduce the processing time for any optimization process. In summary, there exists limited work concerning parallel optimization using lamination parameters. Moreover, to the authors best knowledge, there has been no work (to date) concerning the use of parallel optimization with lamination parameters in a two-level environment to determine structural thicknesses and stacking sequences which satisfy design constraints. Motivated by these shortcomings, the current chapter outlines a conceptual framework to perform the parallel optimization of multi-part laminated composite structures.

6.3 Methodology

Driven by the need for a computationally efficient approach to composite optimization, the following methodology is proposed. Initially, the structure is decomposed into plate elements. A gradient based method is used to minimize the mass of the entire structure (sum of plate elements) subject to local strength and buckling constraints. Local lamination parameters and plate thicknesses are used as design variables. Whilst the objective function and constraints are evaluated on the root processor, the sensitivity analysis is undertaken in parallel using the slave processors. This significantly reduces the computational processing time for each iteration of the optimization. Once the minimum mass, and corresponding thicknesses are minimized, the continuous thicknesses are rounded up to the nearest discrete ply thickness. FEA is then used to ensure that the change in internal loads due to the rounding up is not significantly different from the optimum distribution of internal loads (DIL). It is important to note that this step is conservative. Naturally, if the thickness of one plate is increased, an adjacent or adjoining plate could be decreased and all constraints could still be satisfied. However, to analyse the various combinations of lowering and increasing the thickness of the plate elements would unnecessarily decrease the optimization running time and furthermore steer away from the immediate goal of an efficient approach. Therefore, technically speaking, it is noted that this stage is sub-optimal. At the second level, a combined ACO-DBM approach is used to determine a stacking sequence which satisfies the set of local structural constraints. This is achieved in parallel where each slave processor is used to determine a lay-up of an assigned plate element. Once a lay-up is determined, FEA is used to recalculate the internal loads. If the new load distribution is approximate to the optimal load distribution, the second level is terminated. Otherwise, the second level is reinitiated. The two-level process is summarized in Algorithm 6.1,

Algorithm 6.1

1. **First Level** Decompose the composite structure into plate elements (plates)
2. Assign design variables: 1 thickness, 8 lamination parameters (for a symmetric laminate) to each plate element
3. Setup: Root processor and slave processors. Primary evaluation on root processor - sensitivity analysis undertaken in parallel using slave processors.
4. Local structural constraints (for each plate element) are evaluated using closed form solutions (CFS)
5. **Second Level**: Approximate constraints (Herencia et al. 2008b) using a first or second order Taylor's series expansion if the constraint is not known in closed form. Otherwise use CFS for structural constraints only

6. On each node assign a plate or set of plates if the number of slave processors is less than the number of plate elements

7. For each plate, determine a lay-up which satisfies the local constraints

8. Re-calculate DIL, if the change in the DIL is small (less than a given tolerance, terminate the algorithm, else go to 6) and repeat.

Next, the two-level optimization process in formulized in detail.

6.4 Optimization Formulization

In this section, the continuous and discrete optimization is presented for the design optimization of multi-part laminated composite structures.

6.4.1 Continuous Optimization

At the first level of the optimization, the mass of the wing box is minimized subject to a set of design constraints. The objective function is defined as,

$$M(x) = \sum_{i=1}^{n} a_i b_i t_i \rho \tag{6.1}$$

where n is the number of plates, a_i is the length of the ith plate, b_i is the with of the ith plate, t_i is the thickness of the ith and plate ρ is the density of the material (uniform). Herein, the composite structure under consideration is an idealised wingbox composed of one bay and includes two covers/skins and two side walls. It follows that there are four plates which are denoted as plate elements. The idealised wingbox is shown in Fig. 6.1 and each plate element appropriately labeled. Note, the wingbox has an applied tip load.

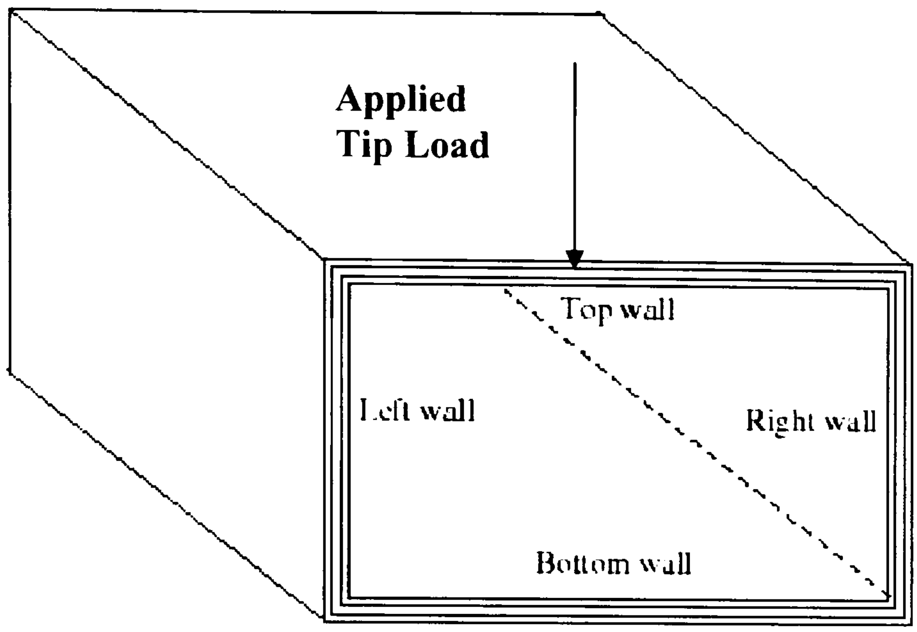

Fig. 6.1: Idealised Composite Wingbox with Applied Load

The design variables, which are continuous, are the local lamination parameters, $(\xi_{1i}^A\ \xi_{2i}^A\ \xi_{3i}^A\ \xi_{4i}^A\ \xi_{1i}^D\ \xi_{2i}^D\ \xi_{3i}^D\ \xi_{4i}^D)$ and panel thickness (t_i) for the ith plate element. Note, the length and width of each panel are fixed. As such only a single finite element model is necessary. If the structural geometry changed, a new finite element model would need to be generated which would drastically increase the computational time for the optimization. The design constraints in the optimization consist of bounds (lower and upper) on the design variables (see Chapter 3) and inequality constraints. The constraints are applied to each plate element. In the current set-up, there are no structural interaction constraints. The set of inequality constraints are,

1. Feasible region of lamination parameters for the selected set of ply orientations assuming each plate is symmetric with respect to the mid-plane and hence does not exhibit any out-of-plane extension coupling
2. Failure strength at laminate level using maximum allowable strains
3. Local critical buckling load determined using closed form solutions.

Note, in order to use the buckling CFS detailed in Chapter 3, one restriction is that the aspect ratio of each plate under consideration must be greater than 3. As such, the plate geometries must be selected accordingly. Details concerning items 1-3 above are given in

Chapter 3. Additionally, each plate is deemed to be simply supported to allow the use of efficient CFS outlined in Chapter 3. When undertaking parallel structural optimization, it is noted that a local change in the design variables will have global effect in the distribution of internal loads (DIL). The sensitivity of change in internal loads with respect to each design variable is performed using the SOL200 module in MSC NASTRAN. Sensitivities with respect to the structural constraints are calculated (using a chain rule approach) as follows,

$$\frac{\partial G_i}{\partial x_j} = \frac{\partial G_i}{\partial x_j} + \frac{\partial G_i}{\partial N_k}\frac{\partial N_k}{\partial x_j} \tag{6.2}$$

The sensitivities $\frac{\partial G_i}{\partial x_j}$ and $\frac{\partial G_i}{\partial N_k}$ are calculated using a standard forward finite difference approximation. On the other hand, $\frac{\partial N_k}{\partial x_j}$, is obtained from the NASTRAN output files. Note, the sensitivity analysis is undertaken using the slave processors. Forward finite differences are chosen as a compromise of efficiency over accuracy (over the central differences method). The sensitivity analysis process naturally lends itself to parallelization as there is no interdependence with the calculations. As such, the computational time of each iteration in the optimization is reduced. Naturally, the reduction in time is not necessarily linear due to the data exchange between root and slave processors. However, having correct setup and implementation (especially dynamic loading where number of plate elements is greater than the number of slave processors) can significantly improve performance to reduce any data bottleneck issues. The continuous optimization terminates when the gradient of the Lagrangian is zero (i.e. the mass of the wingbox is minimized) and hence the change in the DIL is less than a given tolerance. Once the first level is complete (including the rounding post-processing outlined above), the second level is initialized. Note, the geometry of each plate, internal loads and the set of ply orientations is sent to the second level. If linear approximations of the constraints are required, optimal lamination parameters are also passed to the second level of the optimization.

6.4.2 Discrete Optimization

To remind the reader, the goal of the second level is to determine a lay-up for each plate which satisfies the set of design constraints which maintains a stable distribution of the internal loads. To solve the second level problem a combined ACO-DBM, introduced in Chapter 5, is used. The ACO drives the optimizer whilst the DBM is used to perform a local search at each iteration on the current best ant (or randomly selected) in the colony. The ACO-DBM along with the fitness function is described in detail in Chapter 5. Once a stacking sequence for each plate is determined for each plate element, NASTRAN is used to recalculate the DIL using FEA. If the maximum difference between the internal loads over two full consecutive iterations,

$$\max\left|N_{i(t)}^{j} - N_{i(t-1)}^{j}\right| < \varepsilon \qquad \forall i, j \tag{6.3}$$

(where N_i^j is the in-plane load acting in the ith direction on the jth plate) is less than a given tolerance ε, the ACO-DBM is terminated Otherwise the ACO-DBM is again initiated. The procedure continues until the change in internal loads is less than prescribed tolerance. Note, the set-up naturally lends itself to parallelization. This is because the stacking sequence for a given plate element can be determined using a selected slave processor independent of the other processors. The only dependency is when the DIL is calculated which occurs when a lay-up has been determined for each plate element. Next, the optimization and FEA implementation is discussed.

6.5 Implementation

The proposed parallel optimization is undertaken using MATLAB (and the Distributed Computing Toolbox). A cluster of four personal computers (PC) and one root processor (PC) is used. The cluster approach is used at both levels of the optimization. At the first level, each PC in the cluster is used to undertake the sensitivity analysis for a given plate element. This reduces the computational time. At the second level, the ACO-DBM is launched for each plate element. Once a lay-up has been determined which satisfies the local constraints, the root processor is used to calculate the DIL. If the difference between optimal and current DIL is less than a small tolerance, the optimization is terminated and that final design is determined.

As FEA is used to calculate the DIL, the following outlines the FEA modeling using NASTRAN. Each plate element is assumed to be a simply supported symmetrically laminated composite plate. In the FEA modelling, square planar elements (type QUAD4) have been used for the mesh discretization. At the root of the wingbox, all rotations and displacements are fixed. As such, the wingbox is cantilevered. The applied loads, a out-of-plane and a twisting moment, are applied to the tip and transferred to the all points of the contour by means of rigid body elements (type RBE2). The effects of transverse shear are not considered. The material properties are assigned by means of NASTRAN MAT2 option, allowing the inclusion of the in-plane stiffness matrix A. Thus from a given A matrix, FEA calculates the mid-plane strains on each plate as well as the internal loads acting on each plate element. During the optimization, the in-plane stiffness matrix is recalculated and updated in the NASTRAN .bdf input file. Once this process is terminated, the FEA can be performed and the deformations and internal forces N_x, N_y, N_{xy} of each plate element of the laminate are readily calculated. Note, using classical laminate theory for symmetric matrices, the internal loads are a function of the in-plane stiffness (represented by lamination parameters) and mid-plane strains for a given plate element. As such, they are not a function of the laminate stacking sequence but rather the number of each ply orientation in the lay-up.

In the FEA model, it is assumed that middle element of each laminate is representative of the whole wall. As such, the internal loads acting on the middle elements are assumed to be internal loads acting on each of the covers/walls. This assumption is reasonable since no significant difference is observed across the elements of the same wall. Once the internal loads have been determined from the FEA, they can be extracted from the NASTRAN output files and the local design constraints are evaluated accordingly. From the .pch files (NASTRAN output), sensitivities of the mass (objective function) and internal loads with the respect to the design variables are read and loaded into the programme. Local buckling and strength constraints (see Chapter 3) are then evaluated.

Note, global/structural interaction constraints are not considered in the example detailed herein. However, the above framework can be extended in several ways to accommodate more complex structures and structural interaction,

1. More constraints can be added, regarding, for example, bend-twist coupling deformation, useful for aeroelastic purposes or global buckling
2. More complex structures can be analyzed for which there will be no conceptual difference – a decomposition approach can be applied
3. FEA can used to calculate structural interaction, e.g. for buckling of stiffened panels (especially if CFS are not available or accuracy is limiting)
4. Blending between adajcent panels via ply compatability (Liu et al. 2010)

6.6 Conclusions

A method for the parallel optimization of multi-part laminated composite structures has been presented. The method detailed builds upon a two-level approach detailed in Chapters 3-5. At the first level, the mass of each plate element (and hence the composite structure) is minimized subject to local design constraints. The efficiency of the continuous optimization is improved by performing the sensitivity analysis in parallel. It was further observed that the introduction of more processors would further improve this result. Using the approach described herein, the total number of evaluations is identical to the number if the optimization was undertaken on a single terminal. However, by exploiting the parallelization for the sensitivity analysis as well as distributing the second level, significant efficiency savings are made. Furthermore, mass savings are demonstrated using an expanded set of ply orientations. At the second level, a combined ACO-DBM was used to identify stacking sequences which satisfy the set of structural constraints. Each stacking sequence was determined in parallel order to increase the efficiency of the optimization process. Once the set of stacking sequences were determined, the distribution of internal loads were recalculated and the ACO-DBM was repeated to find a new set of stacking sequences which satisfied the newly calculated internal loads. When the change in internal loads was less than a given tolerance, and the minimum reserve factor was greater or equal to zero, the optimization was terminated and the final design was obtained. The conceptual approach presented in this Chapter differs from other approaches in the literature. Liu and Toropov (2010) and Adams et al. (2003) both use a blending approach to allow ply continuity between the adjacent plate elements / panels. However, the approach presented in this thesis is similar to Liu et al. (2003) in the sense that lamination parameters are used as intermediate design variables in the optimization problem. Finally, in Chapter 8, several recommendations are given for future work and enhancements on the aforementioned approach. In the next Chapter, a number of numerical examples concerning composite plates are presented. The motivation behind the numerical examples is to quantify and prove the benefits of the technical advances made in Chapters 2-5.

Chapter 7
Numerical Examples

7.1 Introduction

In the previous four chapters, the optimization approach was articulated. In particular, a two-level optimization was presented for both single laminated composite plates as well as a more general formulation for multi-part laminated composite structures. This chapter encapsulates the analysis as well as the new material derived in this thesis. A two-level optimization approach is used as outlined in Chapter 3. At the first level, the mass of the structure (a composite plate) is minimized subject to a set of local design constraints. These constraints include buckling, strength (allowable laminate strain) and lamination parameter (feasible region) constraints. Note, no further design constraints are considered Once the optimal mass and corresponding thickness(es) is found, they are rounded up to the nearest ply. This represents an increase in the total thickness and hence mass. At the second level, a discrete optimizer (see Chapter 5) is used to solve the constraint satisfaction problem, That is, to determine a lay-up which satisfies the structural constraints only. This is because a stacking sequence will necessarily satisfy the feasible region constraints as it is feasible. For further details, please refer to Chapter 2. In this Chapter, numerical examples concerning the two-level optimization of laminated composite plates are presented.

7.2 Optimization of laminated composite plate

In Chapter 2 it was shown that the full feasible region of lamination parameters was approximated within 2.5% by the set of ply orientations [-90:5:90] degrees. Consequently, this is the largest set considered in this thesis. In the following numerical examples, five sets of ply orientations are used:

1. $[0, 90, \pm 45]$
2. $[0, 90, \pm 45, \pm 30, \pm 60]$
3. $[-90:15:90]$
4. $[-90:7.5:90]$
5. $[-90:5:90]$

The above sets of ply orientations have been selected to represent a diverse range. In particular the sets represent inherently discrete v's semi-continuous sets. For each set of ply orientations, four load cases are considered:

Table 7.1 – Loading Conditions for Simply Supported Rectangular Plate

Load Case	Nx (N/mm)	Ny (N/mm)	Nxy (N/mm)
1	0	0	-800
2	-600	0	300
3	-700	0	0
4	-400	0	-500

The four loading conditions are representative of the minimum and maximum loads placed upon a plate in a typical aircraft wing structure. Finally, three separate plate geometries are considered.

Table 7.2 – Plate Geometry

Geometry Case	*A* (mm)	*b* (mm)	Aspect Ratio
1	1400	200	7
2	1400	300	4.67
3	1400	400	3.5

Note, to allow for the use of closed form solutions (buckling), the assumption is that the plate is long and thin with an aspect ratio greater than three. The geometries selected have been chosen to be representative of those used in a wing section whilst satisfying the above condition. Furthermore, the laminate is assumed to be simply supported on all four sides and symmetric with respect to the mid-plane. Consequently, the laminate does not exhibit any bending extension coupling. This is equivalent to all four coupling lamination parameters being equal to zero. Additionally, shear deformation affects are not considered in the optimization presented herein. Material properties are given in Table 4.2.

7.2.1 Implementation

The first step of the two-level optimization is the calculation of the constraints on the feasible region of lamination parameters for a selected set of ply orientations. The algorithm to achieve this is presented in Chapter 2. Next, the lower and upper bounds on the lamination parameters are calculated. Again, this process is highlighted in Chapter 2. The minimum and maximum thickness, initial lamination parameters (which correspond to the quasi-homogeneous laminate and the centroid of the search space) loading conditions and allowable strains are defined in an input file and loaded into the optimizer. Note in all the numerical examples the allowable strains (at laminate level) are fixed. Increasing the allowable strains may reduce the overall structural mass. The allowable strains (given in microstrains) are shown in Table 7.3,

Table 7.3 – Laminate Allow Strains

AS_{tx}	AS_{ty}	AS_{txy}	AS_{cx}	AS_{cy}	AS_{xy}
3600	3600	7200	-3600	-3600	-7200

where AS_{ij} is the allowable strain and i equals tension (t) or compression (c) and j is the direction of the strain (x, y or xy). Once this step is complete, the continuous optimization begins. The continuous optimization is run using the *fmincon* function of MATLAB

```
[x,fval,exitflag,output,lambda,grad]=fmincon(@objfun,x0,[],[], ...
                        [],[],[LB],[UB],nlc,options);
```

Where '@objfun' is the objective function (mass), *x0* is the initial starting point of the optimization is the lamination parameters which correspond to the quasi-homogenous laminate (all zero) and maximum thickness. The four empty vectors indicated by [], is corresponds the linear equality and inequality constraints. The linear constraints on the feasible region have been placed in the 'nlc' entry of the *fmincon* function. Details of these routines can be found in Appendix B of this thesis.

7.2.2 Continuous Optimization for a Laminated Composite Plate

In Table 7.4, the optimum plate thickness (continuous), lamination parameters and algorithm run details are given. As mentioned, as the laminate is symmetric with respect to the mid-plane all coupling lamination parameters are zero and therefore excluded from Table 7.4.

Table 7.4 – Continuous Optimization Results for Laminated Composite Plates (Strenght, Buckling and Feasible Region Constraints)

M	*T*	ξ_1^A	ξ_2^A	ξ_3^A	ξ_4^A	ξ_1^D	ξ_2^D	ξ_3^D	ξ_4^D	*MLM*	*NI*	*FC*	*PS*	*LC*	*G*
1.956	4.366	-0.106	-0.788	0.413	0	-0.285	-0.429	0.687	0	0.641	34	340	1	1	1
1.843	4.113	-0.291	-0.701	0.225	-0.251	-0.409	-0.569	0.779	-0.745	0.56	20	200	2	1	1
1.838	4.102	-0.21	-0.768	0.279	-0.257	-0.398	-0.537	0.766	-0.791	0.569	19	200	3	1	1
1.831	4.087	-0.251	-0.756	0.264	-0.261	-0.479	-0.425	0.725	-0.847	0.571	16	170	4	1	1
1.828	4.08	-0.27	-0.762	0.268	-0.266	-0.492	-0.422	0.735	-0.862	0.572	17	180	5	1	1
3.837	5.709	-0.11	-0.779	0.615	0	-0.293	-0.414	0.702	0	1.275	41	417	1	1	2
3.586	5.336	-0.408	-0.58	0.416	-0.524	-0.47	-0.494	0.824	-0.843	1.18	19	200	2	1	2
3.584	5.333	-0.301	-0.681	0.442	-0.518	-0.471	-0.475	0.812	-0.852	1.182	26	270	3	1	2
3.574	5.318	-0.235	-0.695	0.448	-0.482	-0.562	-0.271	0.754	-0.916	1.181	26	270	4	1	2
3.568	5.309	-0.283	-0.69	0.459	-0.494	-0.58	-0.235	0.747	-0.949	1.179	19	200	5	1	2
6.195	6.914	-0.135	-0.73	0.681	0	-0.294	-0.412	0.705	0	2.063	32	331	1	1	3
5.783	6.455	-0.408	-0.514	0.502	-0.637	-0.47	-0.477	0.838	-0.848	1.92	26	270	2	1	3
5.781	6.452	-0.419	-0.561	0.525	-0.649	-0.471	-0.468	0.821	-0.863	1.919	25	260	3	1	3
5.767	6.436	-0.267	-0.658	0.532	-0.598	-0.568	-0.256	0.761	-0.925	1.917	30	311	4	1	3
5.757	6.425	-0.239	-0.653	0.526	-0.582	-0.594	-0.199	0.749	-0.963	1.914	26	271	5	1	3
2.217	4.949	0.087	-0.679	0.5	0	0	-0.992	0.14	0	0.651	26	260	1	2	1
2.217	4.948	0.061	-0.66	0.531	-0.043	-0.027	-0.966	0.124	-0.045	0.655	38	390	2	2	1
2.217	4.948	0.062	-0.66	0.539	-0.053	-0.027	-0.966	0.124	-0.046	0.655	38	390	3	2	1
2.211	4.936	0.142	-0.719	0.537	0.09	-0.115	-0.934	0.073	-0.222	0.663	40	410	4	2	1
2.21	4.932	0.115	-0.701	0.541	0.034	-0.085	-0.963	0.094	-0.168	0.659	79	803	5	2	1
4.356	6.482	0.044	-0.813	0.565	0	0	-0.998	0.139	0	1.281	26	270	1	2	2
4.355	6.481	-0.008	-0.767	0.569	-0.123	-0.028	-0.971	0.123	-0.048	1.289	47	480	2	2	2
4.355	6.481	-0.008	-0.767	0.574	-0.127	-0.028	-0.971	0.123	-0.049	1.289	44	450	3	2	2
4.345	6.466	0.09	-0.835	0.561	0.02	-0.116	-0.939	0.072	-0.224	1.305	41	420	4	2	2
4.341	6.46	0.054	-0.813	0.569	-0.049	-0.092	-0.967	0.089	-0.182	1.298	58	590	5	2	2

7.035	7.852	0.046	-0.892	0.361	0	0	-1	0.14	0	1.465	26	289	1	2	3
7.034	7.851	-0.051	-0.829	0.594	-0.163	-0.028	-0.972	0.123	-0.049	2.082	48	490	2	2	3
7.034	7.851	-0.051	-0.829	0.596	-0.166	-0.028	-0.972	0.123	-0.049	2.082	43	440	3	2	3
7.018	7.832	0.048	-0.899	0.582	-0.022	-0.116	-0.94	0.072	-0.224	2.108	42	430	4	2	3
7.011	7.825	0.03	-0.883	0.538	-0.074	-0.094	-0.967	0.088	-0.185	2.098	39	390	5	2	3
2.235	4.988	0.21	-0.58	0	0	0.009	-0.982	0	0	0.74	22	220	1	3	1
2.234	4.988	0.258	-0.6	0	0	0.016	-0.984	0	0	0.741	12	130	2	3	1
2.234	4.988	0.258	-0.6	0	0	0.016	-0.984	0	0	0.741	12	130	3	3	1
2.234	4.987	0.299	-0.608	0	0	0.021	-0.985	0	0	0.74	14	150	4	3	1
2.234	4.987	0.3	-0.608	0	0	0.021	-0.985	0	0	0.74	14	151	5	3	1
4.388	6.53	0.131	-0.737	0	0	0.002	-0.995	0	0	1.46	28	280	1	3	2
4.388	6.53	0.161	-0.751	0	0	0.004	-0.996	0	0	1.46	18	190	2	3	2
4.388	6.53	0.161	-0.751	0	0	0.004	-0.996	0	0	1.46	18	190	3	3	2
4.388	6.53	0.203	-0.762	0	0	0.006	-0.997	0	0	1.46	16	170	4	3	2
4.388	6.53	0.234	-0.765	0	0	0.007	-0.997	0	0	1.46	19	200	5	3	2
7.086	7.909	0.087	-0.826	0	0	0.001	-0.999	0	0	2.361	26	260	1	3	3
7.086	7.909	0.107	-0.835	0	0	0.001	-0.999	0	0	2.361	19	190	2	3	3
7.086	7.909	0.107	-0.835	0	0	0.001	-0.999	0	0	2.361	19	190	3	3	3
7.086	7.909	0.133	-0.844	0	-0.001	0.002	-0.999	0	0	2.361	20	200	4	3	3
7.086	7.909	0.154	-0.849	0	0	0.002	-0.999	0	0	2.361	19	190	5	3	3
2.202	4.916	-0.472	-0.051	0.119	0	-0.176	-0.648	0.477	0	0.494	40	404	1	4	1
2.143	4.784	-0.358	-0.603	0.526	-0.393	-0.359	-0.641	0.348	-0.62	0.498	57	580	2	4	1
2.143	4.784	-0.345	-0.634	0.528	-0.371	-0.36	-0.64	0.347	-0.621	0.498	47	480	3	4	1
2.143	4.783	-0.27	-0.774	0.514	-0.252	-0.345	-0.665	0.358	-0.599	0.496	54	550	4	4	1
2.142	4.782	-0.308	-0.738	0.54	-0.286	-0.325	-0.702	0.371	-0.572	0.494	49	500	5	4	1
4.329	6.442	-0.071	-0.856	0.033	0	-0.176	-0.647	0.478	0	0.758	33	390	1	4	2
4.212	6.268	-0.39	-0.59	0.583	-0.534	-0.36	-0.64	0.347	-0.624	0.979	42	430	2	4	2
4.212	6.268	-0.381	-0.61	0.583	-0.52	-0.36	-0.64	0.346	-0.624	0.979	48	492	3	4	2
4.212	6.267	-0.306	-0.765	0.581	-0.394	-0.345	-0.664	0.357	-0.6	0.976	49	490	4	4	2
4.21	6.266	-0.323	-0.742	0.589	-0.416	-0.325	-0.702	0.371	-0.574	0.971	57	580	5	4	2
6.992	7.804	-0.191	-0.614	0.26	0	-0.176	-0.648	0.477	0	1.568	23	230	1	4	3
6.803	7.593	-0.412	-0.578	0.619	-0.626	-0.361	-0.639	0.346	-0.625	1.581	46	461	2	4	3
6.803	7.593	-0.143	-0.765	-0.009	-0.311	-0.36	-0.64	0.347	-0.624	1.581	28	280	3	4	3
6.803	7.592	-0.139	-0.874	0.12	-0.245	-0.345	-0.664	0.357	-0.6	1.576	30	300	4	4	3
6.801	7.59	-0.344	-0.73	0.625	-0.514	-0.325	-0.702	0.371	-0.574	1.569	49	500	5	4	3

Where , M = Mass (kg), T = Thickness (mm), MLM = Maximum Lagrange Multiplier, NI = Number of iterations, FC = Function Count , PS = Ply Set , LC = Load Case , GC = Geometry Case.

Note, the Lagrange multipliers indicate the change in the objective function if the constraint was relaxed by an infinitesimal amount. Furthermore, the Lagrange multipliers provide an insight into what constraint is driving the optimization and can certainly assist in the design. Lastly as the problem is linear (as discussed in Chapters 2 and 3), at least one constraint should be active (binding). Therefore, the optimum solution should be on the boundary of the feasible region and hence intersecting with at least one constraint. This is identified by a positive Lagrange multiplier and hence all the rows above have a non-zero value in the MLM section.

From Table 7.4, it is shown that the minimum mass (and thickness) is obtained for the largest set of ply orientations. Whilst this is an obvious result, the above table also shows that certain expanded sets of ply orientations have the same optimal mass as a pseudo-continuous. In particular, $0, 90, \pm 45, \pm 30, \pm 60$ degree plies. When the set of ply orientations is $0, 90, \pm 45$, $\xi_4^A, \xi_4^D = 0$. In contrast, when using an expanded set of ply orientations, this constraint is relaxed. As such, lower mass structures are obtained when using an expanded set of ply orientation by utilizing in-plane and flexural anisotropy ($\xi_4^A, \xi_4^D = 0$).Furthermore, it is observed that further mass savings are obtained when the plate aspect ratio of the composite plates is high and the loading conditions are shear driven. In such cases, 60 degree plies ply and fundamental role which is well supported by the literature (Grenestedt 1991) and (Weaver 2006). Finally, it is observed that the optimal vector of lamination parameters does not change significantly between ply orientation set two to five.

Note, the rounded plate thickness (see Chapter 3), plate geometry, loading conditions and sets of ply orientations are passed onto the second level (discrete optimization). In the constraints are not known in closed form, then the optimal lamination parameters are also passed to the second level.

7.2.3 Discrete Optimization

To remind the reader, the objective of the second level is to determine a lay-up (thickness determined at the first level) which satisfies the set of constraints. Formally, this is a constraint satisfaction problem (CSP). At the second level, only the strength (allowable laminate strain) and buckling constraints are considered. This is because the set of feasible region constraints are necessarily satisfied by any lay-up formed of the same set of ply orientations for which the feasible region was calculated in Chapter 2. As the second level design variables are ply orientations and inherently discrete it is not appropriate to use continuous gradient based methods, rather, as outlined in Chapters 4 and 5, several population based meta-heuristic approaches are used.

For each row of Table 7.4, the optimum stacking sequence is determined using

i) ACO-DBM
ii) MPSO
iii) SDGD
iv) GA

The optimization was run such that if the fitness function (see Chapter 4 and 5) is less than a tolerance of 0.009 then the optimization is terminated. Additionally, the maximum number of iterations was 200. If no solution with fitness less than zero is found at the 200 *th* iteration, the algorithm is terminated and the best fitness is retuned. Naturally, this fitness value at the 200 *th* iteration may be above the tolerance level. In Tables 7.5-7.8, the optimum stacking sequence and fitness is given using the above methods.

Table 7.5 Optimal Lay-ups Determined using the ACO-DBM

Example	*Stacking Sequence*	*Fitness*	*NI*	*n*
1	$[90/90/45/45/45/45/45/45/45/45/45/45/-45/-45/-45/-45/45/-45]_{MS}$	0.003	11	18
2	$[60/60/60/60/60/60/45/60/45/60/-60/-45/-60/-60/-60/-45/-45]_{MS}$	0.006	19	17
3	$[60/60/60/60/60/60/60/45/45/45/-45/-45/-45/-60/-45/-45/45]_{MS}$	0.006	124	17
4	$[67.5/67.5/67.5/60/60/60/45/45/45/37.5/-37.5/-52.5/-45/-52.5/-52.5/-45/-75]_{MS}$	0.006	34	17
5	$[65/65/65/65/60/50/60/50/-45/40/40/-45/-50/-70/-50/-50/-45]_{MS}$	0.006	35	17
6	$[45/90/45/45/45/45/45/45/45/45/45/90/45/90/45/-45/45/0/90/0/-45/-45/45]_{S}$	0	6	23
7	$[60/60/60/60/60/60/60/60/60/60/60/60/60/-30/-60/45/-45/-30/45/-45/-60/30]_{MS}$	-0.011	4	22
8	$[75/60/60/60/60/60/60/60/60/45/45/60/60/-45/-45/30/-45/60/45/-45/-30/-30]_{MS}$	0.007	16	22
9	$[67.5/67.5/67.5/67.5/67.5/60/52.5/60/60/60/60/52.5/-45/52.5/37.5/-37.5/-52.5/52.5/45/-37.5/-45/-52.5]_{MS}$	-0.009	13	22
10	$[65/65/65/65/65/65/50/50/55/65/50/60/-40/-40/-45/-55/55/-50/65/65/-50/65]_{MS}$	0.007	11	22
11	$[45/45/45/90/45/45/45/90/90/45/45/90/90/45/90/45/45/45/-45/45/45/45/-45/45/-45/45/45/45]_{S}$	-0.028	2	28
12	$[60/60/60/60/60/60/60/60/60/60/60/60/60/-30/-45/60/-30/45/-60/60/60/90/-45/60/60/90]_{S}$	0.004	3	26
13	$[60/60/60/60/60/60/60/60/60/60/60/60/60/60/-45/60/-45/-15/60/75/-45/45/-45/-60/75]_{S}$	-0.008	3	26
14	$[60/67.5/67.5/67.5/67.5/67.5/67.5/60/52.5/52.5/52.5/52.5/52.5/-45/-30/52.5/67.5/52.5/-22.5/67.5/-45/45/37.5/37.5/-52.5/67.5]_{S}$	0.004	3	26
15	$[65/65/65/65/65/65/65/65/65/65/60/65/-20/-40/-35/50/40/40/-50/65/60/50/45/-55/50/40]_{S}$	0.004	9	26
16	$[45/45/45/-45/-45/45/-45/-45/-45/45/90/0/0/45/0/-45/-45/-45/45/-45]_{S}$	0.008	2	20
17	$[-60/-45/45/45/60/-45/45/45/45/45/-45/45/-45/30/-45/0/-30/90/0/45]_{S}$	-0.015	2	20
18	$[45/45/-45/45/60/-45/-45/45/-30/-45/45/-30/-75/-15/15/-45/-60/-75/30/-30]_{S}$	-0.005	2	20
19	$[-52.5/52.5/45/-45/-45/52.5/45/52.5/45/22.5/-30/52.5/15/37.5/-45/37.5/-15/-75/-45/-67.5]_{S}$	-0.005	2	20
20	$[50/-45/50/-45/50/40/-45/-35/50/25/45/40/-70/30/-30/-30/-55/35/-15/-75]_{S}$	-0.01	2	20

21	*[45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/90/0/0/-45/-45/-45/-45/45/45]$_S$*	0.003	2	26
22	*[45/-60/60/-45/45/-45/-45/45/-45/45/45/45/45/45/-45/-45/45/30/-30/45/0/30/45/-30/-45/-45]$_S$*	0.006	2	26
23	*[45/45/-45/45/-45/-45/45/-45/45/60/-45/-45/45/-45/45/-45/45/45/-30/45/60/75/60/15/0/-30]$_S$*	-0.007	2	26
24	*[-45/45/-45/45/45/45/-45/-45/-45/52.5/45/-45/52.5/52.5/52.5/37.5/37.5/52.5/-45/45/37.5/22.5/-15/-75/-75/-37.5]$_S$*	-0.008	2	26
25	*[55/55/50/-50/-50/-45/50/50/50/45/-40/-35/-40/-30/30/30/30/-45/-30/-45/-35/-45/40/-70/-70/-30]$_S$*	0.008	2	26
26	*[45/-45/45/-45/45/45/-45/45/-45/45/-45/45/-45/45/45/45/-45/-45/-45/45/-45/45/45/-45/90/90/45/-45/0/-45/-45/-45]$_{MS}$*	-0.008	2	32
27	*[60/-60/-45/-45/45/45/45/-45/45/45/-45/45/45/-45/45/-45/45/-45/-45/45/45/45/45/-45/45/45/0/45/45/0/45/45]$_{MS}$*	-0.002	2	32
28	*[45/45/-45/-45/45/-45/60/-45/60/45/-45/45/-45/45/45/-45/-45/45/-45/-45/45/-60/-90/45/15/45/45/45/45/45/30/45]$_{MS}$*	-0.004	2	32
29	*[45/37.5/45/45/-45/-45/-45/-45/-45/45/-45/52.5/-45/-45/52.5/52.5/52.5/52.5/52.5/-45/52.5/52.5/52.5/-45/52.5/30/45/45/22.5/67.5/30/15]$_{MS}$*	-0.007	2	32
30	*[50/45/-45/45/-45/50/-50/50/-50/55/50/-45/-45/50/-45/-45/-45/-45/50/-45/50/50/50/50/50/45/20/20/25/-45/-45/15]$_{MS}$*	-0.017	2	32
31	*[45/-45/45/-45/-45/45/45/-45/-45/45/-45/45/0/45/0/-45/-45/0/0/90]$_S$*	0.005	3	20
32	*[-45/-45/45/-45/45/-45/45/45/45/45/-45/-45/30/30/0/90/-30/0/45/0]$_S$*	0.006	6	20
33	*[-45/45/45/-45/45/45/-45/-45/-45/45/-45/-45/15/15/-45/-30/75/0/0/45]$_S$*	0.006	2	20
34	*[-45/45/-45/-45/45/45/-45/45/45/45/45/-30/-22.5/22.5/-52.5/-22.5/-22.5/7.5/-30/22.5]$_S$*	0.005	7	20
35	*[45/-45/45/40/-45/-45/-45/45/-45/45/40/30/45/-45/15/45/-20/-15/-15/10]$_S$*	0.006	24	20
36	*[-45/-45/-45/45/45/45/-45/45/45/-45/45/45/-45/45/-45/45/45/-45/0/-45/0/90/90/45/45/0/45]$_{MS}$*	-0.035	2	27
37	*[-45/45/45/-45/45/-45/45/-45/-45/45/-45/45/-45/45/-45/45/-45/0/45/45/45/60/0/0/-30/-30/-45]$_{MS}$*	-0.037	2	27

38	*[-45/-45/45/-45/45/45/45/-45/-45/45/45/-45/45/45/-45/-45/45/45/45/-45/30/-30/-75/15/-15/-15/15]*$_{MS}$	-0.042	2	27
39	*[-37.5/-37.5/45/45/45/-45/45/-45/-45/45/45/-45/-45/45/-45/45/-45/45/30/-37.5/-30/-37.5/22.5/7.5/7.5/45/22.5]*$_{MS}$	-0.034	2	27
40	*[40/-40/45/-45/-45/45/45/-45/-45/45/45/-45/-45/25/40/30/-25/-45/-45/-45/10/45/-35/25/-45/25/85]*$_{MS}$	-0.027	2	27
41	*[-45/-45/45/-45/45/-45/45/45/45/45/-45/45/-45/-45/45/-45/-45/45/45/-45/45/45/45/-45/-45/45/0/0/90/-45/-45/90]*$_S$	-0.033	2	32
42	*[-45/-45/45/-45/45/45/-45/-45/45/45/-45/45/45/-45/45/45/45/45/-45/-30/45/-30/-45/-30/-60/-60/90/30/45/60/45/0]*$_S$	-0.029	2	32
43	*[45/-45/-45/45/45/-45/45/-45/45/45/-45/-45/45/-45/-45/45/-45/45/-45/-45/45/45/-30/-45/15/-45/45/-60/-30/-15/-75/30]*$_S$	-0.032	2	32
44	*[45/45/-45/37.5/37.5/-45/-45/-45/45/-45/-45/45/-45/-45/45/45/-45/-45/45/45/-45/-45/-45/45/45/45/-30/30/-15/15/37.5/22.5]*$_S$	-0.029	2	32
45	*[-45/-45/45/45/-45/45/-45/40/-40/-40/45/45/-45/45/45/-45/45/-45/-45/45/45/45/45/45/-45/45/-30/20/15/35/45/5]*$_S$	-0.032	2	32
46	*[0/90/-45/45/45/90/0/0/45/90/45/45/0/90/90/-45/45/0/-45/-45/45/-45/-45/-45/45/-45]*$_{MS}$	-0.304	1	26
47	*[60/60/60/60/60/-45/-45/60/30/45/-60/45/-60/-30/-45/-30/-45/-60/30/-45]*$_{MS}$	-0.048	2	20
48	*[45/45/60/60/60/-45/60/60/-45/-45/45/-45/-60/30/-60/-60/45/-60/-60/-45]*$_{MS}$	-0.039	2	20
49	*[52.5/37.5/-52.5/52.5/-45/45/60/60/-45/60/67.5/67.5/52.5/60/-67.5/60/60/82.5/45/60]*$_{MS}$	-0.025	2	20
50	*[60/55/60/50/50/45/-55/-50/65/-45/-45/-45/-45/60/-45/-45/30/45/20/-45]*$_{MS}$	-0.062	2	20
51	*[45/45/45/45/45/90/90/45/45/45/-45/90/-45/-45/90/-45/-45/-45/-45/-45/-45/45/-45/45/45/-45]*$_S$	-0.04	2	26
52	*[60/60/60/-45/-45/60/60/60/-45/60/60/-45/60/60/60/60/-45/60/-45/60/30/-45/45/30/45/30]*$_{MS}$	-0.074	2	26
53	*[60/60/60/60/60/-45/60/60/-45/-45/-45/60/-45/-45/60/60/60/60/-45/60/15/-45/45/15/60/45]*$_{MS}$	-0.072	2	26
54	*[60/52.5/52.5/52.5/52.5/52.5/-45/-45/60/60/-45/60/-45/60/-45/60/-45/-45/-52.5/-*	-0.066	2	26

	45/-67.5/-60/15/60/37.5/-45]$_{MS}$			
55	*[60/60/55/55/55/-50/60/-45/60/-45/60/60/-45/-45/60/-45/-45/55/55/-45/-45/20/25/45/55/55]$_{MS}$*	-0.073	2	26
56	*[45/45/90/45/45/-45/45/45/90/-45/45/45/45/90/45/45/-45/-45/45/-45/45/45/-45/45/90/90/-45/-45/90/90/-45/45]$_{MS}$*	-0.041	2	32
57	*[60/60/60/60/60/60/-45/60/60/-45/60/60/-45/-45/-45/60/-45/-45/-45/60/-45/-45/60/-45/60/30/45/30/45/-45/45]$_{MS}$*	-0.017	2	31
58	*[60/45/60/-45/60/60/60/-45/60/60/-45/60/-45/60/60/60/-45/60/60/-45/-45/-45/45/45/-75/-45/45/-45/-75/45/-45]$_{MS}$*	-0.01	3	31
59	*[60/60/60/52.5/52.5/-45/52.5/52.5/-45/60/60/-45/60/-45/60/-45/60/-45/-45/60/-45/60/-45/-45/-45/60/-45/37.5/22.5/45/52.5]$_{MS}$*	-0.016	2	31
60	*[60/60/60/55/-45/60/55/-45/55/55/-40/55/55/60/60/-45/-45/-45/60/-45/-45/60/-45/-45/50/15/-45/40/-50/50/-50]$_{MS}$*	0.003	11	18

Table 7.6 Optimal Lay-ups Determined using the MPSO

Example	*Stacking Sequence*	*Fitness*	*NI*	*n*
1	*[45/45/90/45/90/45/45/45/90/45/-45/45/-45/45/45/-45/-45/-45]$_{MS}$*	0.001	50	18
2	*[60/60/60/60/60/60/60/60/-45/-45/45/-60/45/-60/-45/-60/-45]$_{MS}$*	0	44	17
3	*[60/60/60/60/60/60/60/-45/45/45/-60/45/-60/-45/-45/45/-45]$_{MS}$*	0.008	40	17
4	*[60/52.5/67.5/45/60/60/-45/60/52.5/67.5/52.5/-45/-52.5/-52.5/37.5/-45/-52.5]$_{MS}$*	0.007	32	17
5	*[65/55/65/60/60/50/65/45/50/-40/-45/-55/35/40/-50/-50/-45]$_{MS}$*	0.001	20	17
6	*[45/45/90/45/45/45/45/90/45/90/90/45/0/45/0/90/0/-45/90/-45/-45/-45/0]$_{S}$*	0.009	101	23
7	*[60/60/60/60/60/60/60/60/60/60/-30/60/60/-45/90/-45/-60/30/90/-45/30/45]$_{MS}$*	0.002	95	22
8	*[60/60/60/60/45/60/60/60/60/60/60/75/60/45/-45/-45/-45/60/30/-45/-45/0]$_{MS}$*	0.004	82	22
9	*[60/52.5/60/67.5/60/75/60/-90/60/67.5/-37.5/60/67.5/67.5/-45/-30/-60/52.5/22.5/-45/-52.5/-52.5]$_{MS}$*	-0.001	51	22
10	*[70/70/60/60/70/50/60/50/65/65/60/60/-35/-65/40/-40/-35/50/30/-70/-35/-90]$_{MS}$*	0.001	40	22
11	*[90/90/45/45/45/45/90/45/45/90/45/90/0/45/45/45/90/45/-45/90/45/-45/-45/90/0/0/45/90]$_{S}$*	0	10	28

12	*[60/60/60/60/60/60/90/60/60/60/60/60/90/60/60/60/30/60/30/0/60/45/-30/-45/-45/-45]$_S$*	0.007	49	26
13	*[60/60/60/75/75/60/60/75/60/60/60/90/75/60/60/90/45/60/60/-45/-30/60/-60/60/-45/-60]$_S$*	0.001	91	26
14	*[52.5/60/60/67.5/60/60/60/67.5/-82.5/52.5/52.5/52.5/67.5/60/67.5/37.5/75/90/52.5/-15/-37.5/90/-45/-90/-45/45]$_S$*	0.001	33	26
15	*[60/65/65/70/65/60/70/70/60/60/55/-90/60/65/50/-55/-90/-20/-25/-90/-90/50/-40/-40/-40/-80]$_S$*	0.008	83	26
16	*[45/-45/-45/-45/45/45/45/45/-45/45/0/-45/45/90/45/90/-45/45/0/0]$_S$*	-0.008	19	20
17	*[-45/45/60/-45/45/45/-30/30/45/45/60/-30/-30/-60/-60/-30/45/0/-30/0]$_S$*	0.006	17	20
18	*[45/-45/45/-45/-45/60/45/45/90/-30/-45/60/-45/-30/-30/75/30/15/0/-30]$_S$*	-0.005	73	20
19	*[67.5/37.5/-52.5/52.5/-37.5/-45/37.5/45/52.5/-52.5/-45/-37.5/-45/-37.5/30/-7.5/-82.5/-60/30/-15]$_S$*	0.006	50	20
20	*[50/-45/45/50/-75/-45/55/-40/-25/-35/50/45/-45/30/30/90/-90/15/35/-55]$_S$*	0.006	92	20
21	*[-45/45/45/-45/45/-45/45/45/45/-45/-45/45/45/-45/-45/-45/45/90/-45/-45/-45/90/-45/0/90/0]$_S$*	-0.003	8	26
22	*[45/-45/-45/-45/45/60/-60/45/45/-30/-45/45/45/45/-30/45/45/45/45/45/-30/60/0/45/30/60]$_S$*	0.001	110	26
23	*[45/45/-45/45/45/-45/-45/60/45/-45/-45/45/-45/45/75/-45/-60/45/-45/-30/-60/15/90/-90/-75/30]$_S$*	0.001	79	26
24	*[60/-45/-45/-45/45/37.5/52.5/45/52.5/-45/45/-45/60/52.5/-37.5/-45/45/7.5/-67.5/-67.5/-45/-90/90/22.5/60/-37.5]$_S$*	0.001	100	26
25	*[50/-40/-45/45/50/50/45/55/-40/30/50/-50/-50/-50/-45/-90/-45/-70/45/40/-35/-60/-65/10/70/20]$_S$*	0.008	94	26
26	*[-45/45/45/45/45/45/45/-45/-45/-45/45/-45/-45/-45/-45/45/-45/90/45/45/45/-45/-45/-45/45/0/90/90/45/-45/0/45]$_{MS}$*	-0.002	26	32
27	*[-45/45/45/45/45/-45/-45/45/60/30/-45/-60/45/-45/-45/60/-60/45/60/-60/-60/-45/-30/-30/45/60/60/60/30/90/-30/90]$_{MS}$*	0.008	65	32
28	*[45/45/-45/60/-45/60/-45/45/45/-45/-45/60/45/-45/-45/45/-45/30/45/-30/-60/-30/60/-45/-90/0/15/30/45/-60/-45/-90]$_{MS}$*	0.006	48	32

29	*[-52.5/45/-37.5/45/45/-52.5/-45/52.5/67.5/52.5/-45/52.5/52.5/-60/45/52.5/-45/52.5/45/-45/90/-45/60/-30/-45/52.5/7.5/45/-67.5/22.5/82.5/90]$_{MS}$*	0.008	51	32
30	*[50/-45/-40/45/45/-55/45/55/-50/-55/50/50/-40/-45/-40/55/-90/50/50/40/-90/-45/55/50/-45/25/40/90/85/-90/50/-80]$_{MS}$*	0.005	39	32
31	*[45/-45/45/-45/-45/-45/45/0/45/45/45/-45/-45/0/-45/0/45/45/90/0]$_S$*	0.01	200	20
32	*[45/-45/-45/45/-45/45/-30/45/45/-45/-45/45/-60/30/30/-30/0/-60/0/0]$_S$*	0.007	121	20
33	*[45/45/-45/-45/45/-45/45/-45/-45/-45/30/-45/-30/30/75/-15/-90/0/-60/15]$_S$*	0.003	88	20
34	*[-37.5/-45/45/-45/-45/45/45/37.5/45/37.5/45/37.5/-45/-45/45/0/0/7.5/-90/7.5]$_S$*	0.006	75	20
35	*[45/-40/-45/40/45/-40/-40/-45/45/50/40/-60/-45/-35/35/-90/-5/10/90/-10]$_S$*	0	26	20
36	*[45/-45/-45/45/45/-45/45/45/45/-45/45/90/-45/45/0/-45/-45/0/45/90/45/-45/90/90/45/-45/45]$_{MS}$*	0.005	12	27
37	*[30/-45/-60/45/-45/60/-45/45/45/-45/60/-60/30/45/30/-30/-45/45/30/-60/-30/90/-30/30/45/-30/30]$_{MS}$*	0.007	9	27
38	*[45/-45/45/-45/45/-45/-45/-60/45/30/-30/45/-30/-15/30/-60/30/-75/-30/-75/0/0/30/15/-15/90/15]$_{MS}$*	0.007	18	27
39	*[45/-37.5/-60/-37.5/52.5/45/37.5/-45/-45/52.5/60/60/-52.5/37.5/-45/-52.5/0/-52.5/-15/-90/-67.5/-90/-52.5/75/90/-90/7.5]$_{MS}$*	0.007	26	27
40	*[45/45/-50/-50/-45/-45/45/40/0/-40/-55/55/-50/60/65/-35/60/-40/-45/-5/-80/-90/-15/50/-90/50/90]$_{MS}$*	0.008	42	27
41	*[-45/45/-45/45/45/-45/45/-45/45/45/-45/-45/-45/90/45/45/-45/0/-45/90/0/45/-45/45/45/90/0/45/45/90/45/-45]$_S$*	0.002	9	32
42	*[45/30/45/-45/45/45/-60/-45/-45/30/-30/-30/45/45/-45/-45/-60/-45/45/-45/-30/45/-30/-30/-45/30/30/-30/-60/-30/90/0]$_S$*	0.005	40	32
43	*[45/45/45/-45/-60/45/45/-45/-45/-45/-45/45/-45/30/-15/-45/-60/45/45/45/90/-30/0/-60/45/90/-90/-90/-90/-45/-45/90]$_S$*	0.004	69	32
44	*[-45/37.5/-45/-45/-52.5/45/37.5/52.5/37.5/45/37.5/37.5/-45/45/-45/45/67.5/-67.5/-37.5/-45/-30/0/-90/-30/7.5/82.5/-82.5/0/90/-37.5/90/90]$_S$*	0.004	54	32
45	*[45/-40/-45/45/35/-45/-45/40/-50/45/-35/45/-60/-45/-40/40/90/45/-35/-90/55/-90/-60/70/35/80/-45/-45/90/10/35/55]$_S$*	0.007	67	32
46	*[0/-45/90/-45/0/-45/45/90/45/-45/45/45/-45/-45/90/0/-45/45/-45/45/45/45/-45/-45/-*	-0.397	2	26

	45/-45]$_{MS}$			
47	*[60/-30/60/-45/60/90/45/60/90/60/-30/45/-60/45/90/60/45/60/-45/-60]*$_{MS}$	0.003	12	20
48	*[60/45/45/45/-45/60/-45/60/45/45/-60/-30/90/-45/75/-30/-15/-75/-15/60]*$_{MS}$	-0.014	17	20
49	*[67.5/-45/52.5/45/60/75/67.5/22.5/52.5/75/-30/67.5/52.5/-82.5/90/-37.5/-52.5/-37.5/30/-37.5]*$_{MS}$	0.007	16	20
50	*[55/60/-35/55/-90/45/50/90/70/-35/55/-65/60/25/-65/15/40/30/-45/-90]*$_{MS}$	0	15	20
51	*[-45/45/90/45/45/45/90/45/90/45/45/45/-45/-45/90/45/45/45/90/-45/-45/45/45/45/45/45]*$_{S}$	-0.032	6	26
52	*[-30/60/45/60/60/60/-60/-60/45/60/45/-60/30/60/60/60/90/-45/-60/60/90/-30/90/-30/0/-60]*$_{MS}$	-0.006	13	26
53	*[60/45/45/60/-90/-45/-45/60/60/60/-45/45/90/45/90/90/-15/-75/-45/-60/-60/-30/90/-90/-60/-90]*$_{MS}$	0.005	28	26
54	*[45/60/67.5/60/60/-60/-45/37.5/52.5/-90/-22.5/-37.5/45/60/90/-60/90/45/60/-90/-90/90/75/-75/22.5/67.5]*$_{MS}$	0.007	23	26
55	*[50/50/55/-90/50/45/-45/-30/50/70/75/85/60/-35/50/-90/-65/-25/-70/-25/-50/45/-75/-5/-40/30]*$_{MS}$	0.007	24	26
56	*[90/45/45/45/45/45/45/45/90/-45/45/90/90/-45/45/-45/-45/90/0/0/-45/90/90/45/90/45/0/-45/0/90/90/-45]*$_{MS}$	-0.013	11	32
57	*[60/60/60/60/60/60/-45/60/60/-60/60/60/-45/60/-45/0/-45/60/-45/-45/-30/-45/-30/-45/-60/60/-45/90/45/30/-60]*$_{MS}$	0.006	64	31
58	*[45/60/60/60/60/-45/60/60/-45/-30/60/60/60/-45/60/-90/-45/-45/60/-45/-90/60/-60/60/45/45/-90/60/-90/-45/75]*$_{MS}$	0.006	52	31
59	*[67.5/52.5/-37.5/60/60/-45/45/60/60/60/-45/-52.5/60/45/-22.5/60/60/67.5/60/-37.5/52.5/60/-90/60/-45/52.5/60/90/67.5/-82.5/-90]*$_{MS}$	0.008	17	31
60	*[55/50/-45/55/65/55/-45/45/60/55/65/55/-50/60/60/-45/70/45/-40/-85/-90/-35/60/60/-10/-90/-90/90/-75/40/-85]*$_{MS}$	0.008	50	31

Table 7.7 Optimal Lay-ups Determined using the SDGD

Example	***Stacking Sequence***	***Fitness***	***NI***	***N***

1	*[45/90/45/90/45/45/45/45/45/45/45/45/-45/45/-45/-45/-45/-45]$_{MS}$*	-0.001	98	18
2	*[60/60/60/60/60/60/60/45/45/-45/-45/45/-45/-45/-45/-60/45]$_{MS}$*	0.009	114	17
3	*[60/60/60/60/60/60/60/45/45/-45/45/-45/-45/-60/-45/-45/45]$_{MS}$*	0.007	114	17
4	*[67.5/60/60/60/60/52.5/45/67.5/52.5/45/-52.5/-45/-52.5/-52.5/-45/-45/-52.5]$_{MS}$*	0.004	73	17
5	*[65/65/65/60/60/50/40/45/60/45/-50/40/-45/-45/-45/-50/-50]$_{MS}$*	0.006	90	17
6	*[90/45/45/45/45/90/90/45/45/45/45/45/-45/-45/45/45/45/45/90/45/-45/45/0]$_S$*	0.006	14	23
7	*[60/60/60/60/60/60/60/60/60/60/-30/60/45/60/-45/-30/-60/60/30/-60/-45/-45]$_{MS}$*	-0.003	10	22
8	*[60/60/60/60/60/60/75/60/60/45/-45/45/60/60/60/45/45/-45/45/-45/-45/-60]$_{MS}$*	0.005	54	22
9	*[67.5/60/60/67.5/67.5/67.5/60/52.5/37.5/60/67.5/-45/60/45/-52.5/52.5/45/45/52.5/-37.5/-52.5/-52.5]$_{MS}$*	0.007	57	22
10	*[60/60/65/60/60/65/65/60/50/60/-35/55/45/55/-80/65/45/40/-50/-45/-45/-50]$_{MS}$*	-0.005	54	22
11	*[45/90/45/45/45/45/45/45/90/90/45/45/45/90/90/-45/-45/90/-45/0/0/0/0/-45/90/45/90/45]$_S$*	0.007	8	28
12	*[60/60/60/60/60/60/60/60/60/60/60/60/90/45/0/30/60/60/90/-30/-30/30/45/-30/-60/-60]$_S$*	0.001	9	26
13	*[75/60/60/60/60/60/60/60/45/45/60/60/45/60/60/-30/45/60/45/45/-90/45/90/-45/-30/-45]S*	0.008	31	26
14	*[60/52.5/60/75/60/60/67.5/60/60/52.5/67.5/67.5/67.5/67.5/-7.5/67.5/67.5/-52.5/-60/15/-52.5/-67.5/60/-90/52.5/-60]$_S$*	0.004	13	26
15	*[60/60/60/60/60/65/60/65/65/65/70/65/80/55/-15/-65/85/-85/80/-90/-55/10/40/-65/-65/-50]$_S$*	0.002	11	26
16	*[45/45/-45/-45/-45/45/45/-45/-45/45/45/-45/0/0/45/0/45/90/-45/0]$_S$*	-0.005	5	20
17	*[-45/45/-45/60/-45/60/45/45/45/-60/-45/60/0/0/0/90/45/60/0/0]$_S$*	0.001	6	20
18	*[45/45/-45/45/-45/45/-45/-45/-75/30/-45/-90/45/15/-75/-75/15/15/15/-30]$_S$*	0.007	9	20
19	*[-52.5/-45/52.5/30/-45/52.5/52.5/52.5/45/45/-67.5/22.5/-30/37.5/45/37.5/-60/-15/-45/7.5]$_S$*	-0.001	30	20
20	*[55/-50/50/-50/45/35/-55/45/50/-45/-45/-80/-5/-25/-25/-20/-20/-15/30/-20]$_S$*	0	7	20
21	*[45/45/45/-45/-45/-45/-45/45/45/45/45/-45/45/90/-45/-45/-45/-45/45/45/0/45/90/90/45/0]$_S$*	0.001	10	26
22	*[-45/60/-45/-45/45/-45/45/45/45/60/45/45/60/-45/-30/30/-45/90/45/30/60/-*	0.007	7	26

	30/0/60/45/-30]$_S$			
23	*[-45/-45/45/60/45/45/-45/-45/45/30/-45/-45/45/45/45/45/45/60/75/-45/45/-90/30/15/60/60]$_S$*	0.004	25	26
24	*[45/45/-45/-45/-37.5/52.5/45/-52.5/45/60/-45/37.5/-45/37.5/-52.5/-52.5/52.5/7.5/45/52.5/-52.5/-90/90/30/75/60]$_S$*	0.006	10	26
25	*[45/50/-45/-45/50/-45/-45/45/45/-45/-40/35/65/70/50/55/-25/-45/90/15/-70/20/-40/25/5/90]$_S$*	0.008	10	26
26	*[-45/45/45/-45/45/45/45/-45/45/-45/45/-45/-45/45/45/-45/45/-45/45/90/90/-45/-45/0/90/0/-45/0/90/-45/0/90]$_{MS}$*	0.004	13	32
27	*[45/60/-45/60/45/-45/-45/45/-45/-45/-45/-45/45/45/45/60/-60/-45/45/45/-30/-45/-30/90/90/90/0/-30/-45/90/45/-60]$_{MS}$*	0.005	11	32
28	*[-45/45/45/-45/45/-45/-45/45/-45/60/45/45/-30/45/45/-45/60/60/-45/-30/30/90/90/60/75/15/-60/30/90/-45/45/-45]$_{MS}$*	0.008	10	32

29	*[45/52.5/52.5/-45/-45/-45/45/45/-45/-52.5/-45/52.5/45/67.5/45/45/-60/-67.5/-45/-52.5/-15/-37.5/82.5/22.5/60/37.5/15/45/90/-30/-67.5/-82.5]*$_{MS}$	0.005	13	32
30	*[55/50/50/-45/45/-45/40/45/-45/-45/-40/-45/25/50/35/-45/-45/-50/50/-55/65/-50/-90/-70/35/60/-70/-65/65/80/75/-80]*$_{MS}$	0.006	14	32
31	*[45/-45/45/-45/45/-45/-45/-45/45/-45/45/45/45/0/0/45/45/0/90/0]*$_{S}$	0.003	32	20
32	*[-45/-45/45/-45/45/45/-45/45/30/-45/45/30/-45/60/-30/30/0/-30/0/0]*$_{S}$	0.007	40	20
33	*[45/-45/-45/45/45/-45/45/-45/45/-45/-30/30/-30/-60/-15/30/-75/45/15/0]*$_{S}$	0.007	38	20
34	*[45/45/-45/-45/45/-45/-37.5/45/-45/45/-45/-37.5/-15/22.5/-67.5/-45/22.5/-22.5/15/22.5]*$_{S}$	0.007	26	20
35	*[45/-45/-45/45/45/45/-45/-45/-45/45/-30/-40/45/15/-25/-70/-5/-40/30/5]*$_{S}$	0.005	56	20
36	*[45/-45/45/-45/-45/-45/45/45/45/-45/45/-45/90/45/0/-45/0/-45/0/-45/45/90/0/0/90/0/0]*$_{MS}$	-0.005	11	27
37	*[45/-45/-45/-45/30/60/45/-45/45/-60/60/-60/-45/30/45/-30/30/-60/90/45/45/-30/-45/-30/0/-60/0]*$_{MS}$	0	6	27
38	*[45/-45/45/-45/-45/45/30/-45/-45/45/-45/45/-15/-15/45/60/-30/45/75/15/-75/-30/45/30/-30/75/-60]*$_{MS}$	-0.003	10	27
39	*[-45/45/-52.5/-45/-30/-45/37.5/52.5/45/52.5/67.5/-45/45/37.5/45/22.5/-75/0/-7.5/7.5/60/-30/-82.5/52.5/-75/-75/45]*$_{MS}$	0.007	6	27
40	*[40/-45/45/-45/40/-45/-45/-45/40/5/65/-35/-60/50/40/-45/70/80/-45/80/55/-90/-50/-30/-70/-15/-75]*$_{MS}$	0.008	7	27
41	*[45/-45/-45/45/45/-45/-45/45/-45/45/-45/45/-45/45/45/45/-45/-45/90/90/0/90/0/90/-45/90/90/0/90/0/45/-45]*$_{S}$	0	9	32
42	*[-45/45/60/-45/45/-45/-45/-45/-45/45/45/45/-30/-30/30/60/30/60/-30/30/60/60/-30/-60/-45/-30/45/-45/60/45/-45/45]*$_{S}$	-0.001	6	32
43	*[45/45/45/-45/-30/-45/-60/45/45/-30/-45/-45/-60/45/-45/-45/45/60/15/45/-30/90/-45/-30/45/90/-30/-30/15/45/-15/30]*$_{S}$	0.005	12	32
44	*[45/-45/45/45/-45/-45/-45/45/-45/-30/-30/45/-37.5/45/-45/52.5/60/-15/90/-45/37.5/52.5/0/-15/67.5/90/75/7.5/-67.5/-15/7.5/75]*$_{S}$	0.005	10	32
45	*[45/-45/-40/55/40/45/45/-45/-45/-30/-30/-60/-45/-45/50/60/40/45/-45/40/90/-25/40/-70/80/-5/-85/5/20/50/65/-85]*$_{S}$	0.004	12	32

46	*[90/45/90/-45/45/-45/45/0/45/-45/-45/0/-45/45/45/90/90/0/45/90/-45/90/45/45/-45/-45]$_{MS}$*	-0.344	1	26
47	*[60/60/60/30/-45/60/-60/60/60/-30/60/-30/-60/90/45/-60/90/90/90/45]$_{MS}$*	-0.021	4	20
48	*[60/-45/60/45/90/75/60/60/60/-30/60/30/0/-90/-45/-45/-45/45/15/-45]$_{MS}$*	0.001	5	20
49	*[67.5/45/67.5/45/-30/-45/52.5/75/60/37.5/-67.5/52.5/-37.5/-52.5/-7.5/37.5/60/15/75/52.5]$_{MS}$*	-0.007	4	20
50	*[55/65/50/55/-45/55/55/-80/-55/45/10/-65/-5/-45/70/-10/40/-55/50/35]MS*	-0.009	5	20
51	*[45/45/45/45/45/90/90/90/-45/45/-45/90/90/90/-45/45/0/0/0/90/-45/45/-45/45/-45/-45]$_S$*	-0.002	6	26
52	*[60/60/60/60/60/60/-30/-30/60/90/-30/60/-45/-60/45/0/0/90/45/45/0/-60/30/45/45/30]$_{MS}$*	0.002	4	26
53	*[45/60/60/-30/-60/60/60/60/60/-75/-30/60/60/75/-60/45/75/-75/-90/75/90/45/-30/60/-45/-90]$_{MS}$*	-0.008	6	26
54	*[60/52.5/60/52.5/60/-30/45/52.5/-52.5/-22.5/-37.5/82.5/-82.5/60/22.5/-45/60/37.5/82.5/67.5/15/-22.5/-67.5/-7.5/45/-7.5]MS*	-0.002	7	26
55	*[70/50/60/60/-45/55/-40/55/80/55/50/-40/70/-65/0/50/65/-80/-15/15/-80/-70/80/80/40/-25]$_{MS}$*	-0.012	7	26
56	*[45/45/45/45/90/-45/45/45/-45/90/-45/45/45/45/90/45/45/0/45/0/0/90/90/45/-45/90/45/0/90/45/-45/90]$_{MS}$*	-0.01	5	32
57	*[60/60/60/-45/60/-45/60/60/60/60/60/-45/-45/-60/60/45/-30/45/30/60/-60/-30/0/45/30/-60/-30/90/60/90/-30]$_{MS}$*	0.006	8	31
58	*[60/60/60/60/-45/60/60/60/45/-45/-45/-30/-60/60/45/-45/60/60/60/-45/75/75/-15/45/90/30/90/-30/-75/75/75]$_{MS}$*	0.004	14	31
59	*[60/60/52.5/67.5/52.5/52.5/60/-37.5/52.5/-37.5/-45/-52.5/52.5/60/-67.5/-45/-45/52.5/67.5/90/-52.5/45/-75/30/67.5/30/30/75/22.5/-75/52.5]$_{MS}$*	0.008	11	31
60	*[55/60/65/65/-50/55/55/-45/50/60/-50/65/55/55/-45/55/40/-25/-40/-20/-45/85/80/-10/80/-45/-85/-45/0/-25/-30]$_{MS}$*	0.007	12	31

Table 7.8 Optimal Lay-ups Determined using the GA

Example	*Stacking Sequence*	*Fitness*	*NI*	*n*
1	*[90/45/45/90/45/45/45/45/45/45/45/45/-45/-45/45/-45/-45/-45]* $_{MS}$	0.00	45.00	18.00
2	*[60/60/60/60/60/60/-30/60/45/30/-60/-45/45/-60/-45/-60/-45]* $_{MS}$	0.03	200.00	17.00
3	*[60/60/60/60/60/45/-60/60/60/45/-60/60/-45/-45/-45/-45/-45]* $_{MS}$	0.04	200.00	17.00
4	*[67.5/60/60/52.5/52.5/67.5/37.5/67.5/30/37.5/-52.5/-52.5/45/-45/-45/-45/-67.5]* $_{MS}$	0.03	200.00	17.00
5	*[50/50/60/60/50/65/55/70/50/-40/-45/-50/-40/-30/40/-60/35]* $_{MS}$	0.05	200.00	17.00
6	*[45/45/45/45/45/90/90/90/45/90/45/90/45/-45/-45/45/45/90/45/0/-45/45/-45]* $_S$	0.00	30.00	23.00
7	*[60/60/60/60/60/60/60/45/60/45/60/60/-30/45/60/-60/30/45/-60/-30/-60/-30]* $_{MS}$	0.01	198.00	22.00
8	*[60/60/60/60/60/75/60/45/60/75/-30/60/45/60/-30/45/-75/-45/-45/45/-45/15]* $_{MS}$	0.01	176.00	22.00
9	*[52.5/67.5/60/60/67.5/67.5/67.5/67.5/52.5/75/52.5/45/52.5/-60/75/-52.5/-90/-45/-52.5/-37.5/30/-37.5]* $_{MS}$	0.01	53.00	22.00
10	*[65/60/60/80/65/45/60/65/40/50/65/60/-35/70/45/25/-85/-45/-65/-25/-40/-55]* $_{MS}$	0.02	200.00	22.00
11	*[45/90/45/45/45/45/45/90/90/45/45/90/90/45/45/90/90/-45/0/0/90/45/45/-45/90/-45/90/-45]* $_S$	-0.02	7.00	28.00
12	*[60/60/60/60/60/60/60/60/60/60/45/45/60/60/45/60/60/60/0/0/-45/-30/-45/-45/-45/-60]* $_S$	0.00	100.00	26.00
13	*[60/60/60/60/60/45/60/75/75/75/60/60/75/60/45/30/75/75/45/-30/-15/-60/-60/-45/75/-75]* $_S$	0.00	134.00	26.00
14	*[52.5/67.5/52.5/75/67.5/52.5/60/67.5/67.5/67.5/60/52.5/67.5/60/52.5/60/45/45/90/-60/-52.5/-52.5/75/30/0/-37.5]* $_S$	0.01	182.00	26.00
15	*[75/60/65/60/55/65/65/80/50/50/50/70/45/60/65/75/70/50/45/-45/90/-55/-45/-20/-30/45]* $_S$	0.01	184.00	26.00
16	*[-45/45/45/45/45/45/-45/-45/-45/45/0/45/45/-45/0/-45/-45/45/0/90]* $_S$	0.00	13.00	20.00
17	*[45/-45/-45/45/60/-60/60/45/-60/-30/45/-45/45/30/90/0/-60/-60/0/45]* $_S$	0.00	4.00	20.00
18	*[60/-45/-45/45/-45/30/60/45/45/45/-30/-90/-60/-45/-15/45/0/30/0/15]* $_S$	0.01	31.00	20.00
19	*[37.5/-52.5/-45/60/52.5/45/37.5/-45/-52.5/60/52.5/30/75/-90/-60/-22.5/-7.5/52.5/-15/-22.5]* $_S$	0.01	37.00	20.00
20	*[45/-50/-55/-40/45/40/70/50/40/30/-45/-55/20/65/30/75/80/5/-50/-85]* $_S$	0.01	25.00	20.00
21	*[45/-45/-45/45/-45/45/45/45/-45/45/-45/45/45/-45/45/-45/-45/0/90/90/90/90/0/90/90/45]* $_S$	0.01	13.00	26.00

22	*[45/45/-45/-45/45/-60/45/-45/45/45/45/-30/45/60/-45/30/-60/-45/-45/-30/-60/30/-45/-45/-30/0]$_S$*	0.01	80.00	26.00
23	*[45/45/-45/-45/45/45/45/-45/45/-45/45/45/-60/-60/-30/-45/45/-30/-60/-60/0/-15/45/-15/45/-30]$_S$*	0.01	50.00	26.00
24	*[45/-52.5/-45/45/52.5/52.5/-30/45/-52.5/-52.5/52.5/52.5/-45/37.5/60/-52.5/-37.5/-22.5/37.5/30/-22.5/60/0/45/90/-52.5]$_S$*	0.01	58.00	26.00
25	*[35/55/45/50/55/-50/-50/-45/-50/-50/50/-45/-45/-45/-30/-15/50/-45/-40/20/-15/-35/55/25/35/40]$_S$*	0.01	200.00	26.00
26	*[-45/45/-45/45/45/45/-45/45/45/-45/45/-45/45/-45/-45/-45/-45/45/90/90/45/90/-45/-45/0/-45/45/-45/-45/0/45/0]$_{MS}$*	0.01	10.00	32.00
27	*[-45/-45/45/-45/60/45/45/-60/60/60/45/45/45/-45/45/45/-45/45/-45/-30/45/30/-30/-30/-60/45/-60/0/90/-30/-60/0]$_{MS}$*	0.01	29.00	32.00
28	*[45/45/-45/-45/60/45/45/-45/-45/-45/-45/45/-45/-45/60/-45/45/-30/-30/45/60/45/45/-60/-30/60/-90/60/-30/30/75/-45]$_{MS}$*	0.00	68.00	32.00
29	*[52.5/-45/-52.5/-45/-45/60/45/37.5/-37.5/60/45/45/-37.5/60/60/45/-45/-52.5/45/60/-37.5/52.5/60/-60/82.5/30/-67.5/-52.5/30/45/90/-37.5]$_{MS}$*	0.01	76.00	32.00
30	*[-50/-40/50/45/45/45/45/-45/-40/50/55/50/60/40/-45/-35/-60/60/-10/-55/-55/-35/45/-55/65/50/40/-80/-15/65/65/-25]$_{MS}$*	0.01	112.00	32.00
31	*[45/45/45/-45/-45/-45/-45/45/-45/-45/45/-45/-45/90/45/0/0/0/45/0]$_S$*	0.00	25.00	20.00
32	*[45/-45/-45/45/-45/45/-45/45/30/-45/-30/-30/-45/45/45/30/0/0/45/0]$_S$*	0.01	200.00	20.00
33	*[-45/-45/45/45/-45/45/45/-45/-45/45/-15/45/-30/30/30/-45/60/15/15/0]$_S$*	0.01	200.00	20.00
34	*[45/37.5/-52.5/-30/52.5/-45/52.5/-45/-52.5/-45/37.5/-45/-37.5/37.5/-52.5/30/15/75/0/-7.5]$_S$*	0.02	200.00	20.00
35	*[45/-45/-55/40/45/45/-35/-40/40/-55/-45/-45/40/40/-15/-40/-20/5/25/-15]$_S$*	0.01	200.00	20.00
36	*[-45/-45/45/45/45/45/-45/90/-45/45/45/45/45/45/-45/-45/0/90/-45/45/45/-45/0/45/-45/45/0]$_{MS}$*	0.01	4.00	27.00
37	*[45/-45/45/30/-45/-45/45/-30/30/-45/-60/30/60/-60/30/-60/30/-45/-30/-45/-45/90/60/0/90/-60/-60]$_{MS}$*	0.01	11.00	27.00
38	*[45/-30/-45/-45/45/-45/45/60/45/-45/-30/-45/-30/-30/30/-30/60/45/-60/-15/30/-90/30/45/-45/-75/-75]$_{MS}$*	0.00	18.00	27.00

39	*[37.5/-45/52.5/52.5/-37.5/45/-52.5/-45/-52.5/30/60/-52.5/-60/15/52.5/-37.5/52.5/-7.5/-37.5/75/90/30/-15/-30/-60/45/-37.5]*$_{MS}$	0.01	42.00	27.00
40	*[45/-45/-35/50/50/50/-65/-50/-50/55/45/-25/-60/40/-45/50/-65/-55/-55/55/-45/20/-35/20/5/10/-55]*$_{MS}$	0.00	43.00	27.00
41	*[-45/-45/-45/45/45/-45/45/45/-45/0/-45/45/45/45/45/45/45/-45/-45/0/-45/45/-45/45/45/45/45/45/0/90/90/0]*$_{S}$	-0.01	3.00	32.00
42	*[-45/45/-60/45/-45/-45/30/60/45/-45/45/30/-45/30/30/-45/-60/60/30/-60/45/45/30/-60/45/45/30/-60/30/90/-45/0]*$_{S}$	0.01	29.00	32.00
43	*[45/-45/45/45/45/45/-45/-30/-45/-45/-45/45/-30/45/-30/-30/75/-60/60/45/60/60/-45/-30/-15/45/75/0/15/-15/-60/15]*$_{S}$	0.01	31.00	32.00
44	*[-52.5/45/-45/-45/45/45/45/-37.5/45/37.5/-52.5/-67.5/37.5/-67.5/45/22.5/52.5/-60/37.5/37.5/-52.5/-22.5/-60/30/90/45/-7.5/-82.5/30/15/52.5/75]*$_{S}$	0.01	29.00	32.00
45	*[35/-50/45/40/-50/-35/60/-55/35/40/50/-45/45/40/-60/-30/-60/-30/-55/-65/35/50/-50/-30/-65/-45/40/35/35/-70/-45/70]*$_{S}$	0.01	92.00	32.00
46	*[-45/0/45/0/-45/0/-45/90/90/-45/45/-45/45/0/45/45/0/-45/90/45/-45/-45/45/-45/-45/0]*$_{MS}$	-0.33	1.00	26.00
47	*[60/60/60/60/-60/90/45/-60/-30/45/30/-45/-45/45/30/30/0/30/0/90]*$_{MS}$	0.00	9.00	20.00
48	*[60/60/60/75/60/30/-60/-45/45/-45/-30/15/-45/75/-60/-45/0/0/-15/-15]*$_{MS}$	0.00	17.00	20.00
49	*[52.5/-37.5/60/37.5/52.5/82.5/37.5/-37.5/45/-60/67.5/52.5/-37.5/75/45/52.5/-82.5/-82.5/90/52.5]*$_{MS}$	0.00	13.00	20.00
50	*[60/-55/45/50/50/-25/55/65/60/50/-70/15/80/65/-50/-20/50/10/-90/-60]*$_{MS}$	-0.01	11.00	20.00
51	*[-45/45/45/45/90/45/-45/90/90/90/45/45/45/45/90/-45/45/-45/0/90/0/45/-45/0/45/90]*$_{S}$	0.00	3.00	26.00
52	*[60/-45/45/45/60/45/90/60/45/60/45/30/-60/-60/-45/-30/90/-45/-45/45/-60/60/90/90/-30/-45]*$_{MS}$	0.01	10.00	26.00
53	*[60/-45/75/60/45/60/45/-45/75/45/45/-15/-75/-60/60/75/-15/-30/-45/75/45/-60/-90/-75/0/90]*$_{MS}$	0.01	16.00	26.00
54	*[-37.5/60/67.5/60/52.5/-30/60/-30/60/60/60/22.5/52.5/90/30/90/52.5/-75/-82.5/-37.5/15/45/-82.5/90/75/-37.5]*$_{MS}$	0.00	23.00	26.00
55	*[65/45/-45/60/60/40/-75/60/35/-25/60/-55/-70/70/70/55/55/65/45/-90/-*	0.01	14.00	26.00

	75/40/75/85/10/-45]$_{MS}$			
56	*[45/90/90/45/45/-45/90/45/-45/45/90/45/90/45/45/-45/-45/-45/90/45/45/90/-45/45/45/0/90/45/-45/90/45/0]$_{MS}$*	0.01	1.00	32.00
57	*[60/60/60/60/-30/-60/60/60/45/60/-45/60/60/60/-30/60/-45/45/60/-60/-45/-45/60/-45/-30/-60/-45/45/45/-45/-60]$_{MS}$*	0.01	132.00	31.00
58	*[60/60/45/60/60/60/-30/60/60/-45/-45/75/-45/-45/-60/-45/60/45/45/60/60/75/75/60/-60/60/-30/-45/-90/-90/75]$_{MS}$*	0.01	89.00	31.00
59	*[60/-60/60/-45/52.5/-37.5/67.5/52.5/52.5/60/52.5/52.5/60/45/60/75/-45/52.5/45/52.5/67.5/45/-30/-52.5/67.5/75/60/-37.5/52.5/-37.5/-60]$_{MS}$*	0.01	170.00	31.00
60	*[-45/70/60/-40/50/60/40/55/55/65/60/60/-55/60/-45/-55/65/60/55/60/60/45/-40/55/55/65/5/60/80/65/60]$_{MS}$*	0.01	153.00	31.00

7.2.4 Discussion of Results

The first level results are summarized in Table 7.9.

Table 7.9 Continuous Optimization Results Summary

Load Case	*Geometry Case*	*% Mass Saving Between PS_1 and PS_5*	*% Saving Between PS_1 and PS_2*
1	1	6.5	5.7
1	2	7	6.5
1	3	7.1	6.7
2	1	0.3	0
2	2	0.3	0
2	3	0.3	0
3	1	0	0
3	2	0	0
3	3	0	0
4	1	0	0
4	2	1.7	1.2
4	2	1.7	1.2

From Table 7.9, several conclusions can be made. Firstly, the mass savings are achieved where the aspect ratio is high and the plate is shear loaded and using an expanded set of ply orientations. Generally speaking, if the loading conditions are shear dominated, using an expanded set of ply orientations will yield a lower objective function. An additional observation is that the optimum set of ply orientations for normally loaded plates (or normally dominant) is indeed the restricted set of $[0,90,\pm 45]$ degree plies thus supporting current industrial practices. However, in the realm of multi-part laminated composite design, the internal loads change during the optimization, thus it is difficult at best to determine a priori the optimal set of ply orientations to choose from. Based upon the above results the expanded set of ply orientations $[0,90,\pm 45,\pm 30,\pm 60]$ give the designer a compromise over allowing more efficient designs, reducing the potential size of the set of feasible ply orientations and from a practical perspective may reduce manufacturing complexity (relative to a continuous set of ply orientations).

In Tables 7.5-7.8, the four second level algorithms show varied performance. The results clearly show that the ACO-DBM outperformed the other algorithms. In particular, showing up to fourty percent savings in the average number of iterations compared to the methods outlined in Chapter 4. The GA had the worst (relative) performance (supported by the analysis and results of Chapter 4). This is indicated with a high number of iterations and (relatively) high fitness of optimal solutions. Despite, each of the four methods found a good solution by 200 iterations. This is indicated by a maximum fitness across all 240 examples of 0.009 which is very close to zero. As such, the GA should not be dismissed entirely.

It is observed that it is generally harder to solve the second level problem for smaller sets of ply orientations. This is because at the first level, the optimization assumes continuous ply thickness. At the second level, ply thickness is discrete. As the number of ply orientations increases, so does the number of possible lay-ups. The precise number is m^n where m is the number of unique ply orientations and n is the number of plies (which is half the total number of plies as the examples are restricted to symmetric laminates only). As the number of lay-ups increases, so do the range of feasible lamination parameters, therefore making it more feasible to move closer to the vector of optimal lamination parameters or at least a feasible solution to the CSP. At the first level of the optimization, fixed ply orientations and continuous ply thickness is assumed (in lamination parameter space). At the second level it should be possible to match the optimum lamination parameters by having a fixed ply thickness, but having a continuous set of ply orientations. Mathematically, the two problems are equivalent. In practice, fixed ply orientations are necessary at both levels leading to the above observation that it is easier to match optimum lamination parameters (say, in a least squares sense), when a larger set of ply orientations is considered. In sum, there is a relationship between the number of plies (laminate thickness) and the range of feasible lamination parameters. However, the inclusion of such a relationship may transform the problem to a non-convex one which is naturally undesirable. To remind the reader, the optimization presented in this thesis is convex, for for multi-part composites which entail multiple thicknesses, the resulting problem is generally non-convex.

In the next section, numerical examples using the DBM are given. In particular, the DBM is used to maximize the buckling reserve factor constraint for a determined minimum thickness.

7.3 Numerical Examples Using the Direct Branching Method

In this sub-section, numerical examples concerning the DBM are presented. Again, the two-level optimization framework is utilized. At the first level, the mass of the laminated composite plate is minimized subject to buckling and lamination parameter constraints only. At the second level, the DBM is used to maximize the constraint reserve factor of the laminate buckling constraint using an initial laminate idealization detailed in Chapter 5.

7.3.1 – Continuous Optimization

In Table 7.10, optimal lamination parameters, thickness and algorithm details are provided.

Table 7.10 – Continuous Optimization Using the Buckling and Feasible Region Constraints

M	***T***	ξ_1^A	ξ_2^A	ξ_3^A	ξ_4^A	ξ_1^D	ξ_2^D	ξ_3^D	ξ_4^D	***MLM***	***NI***	***FC***	***PS***	***LC***	***G***
1.951	4.355	-0.112	-0.776	0.886	0	-0.294	-0.412	0.706	0	0.65	28	280	1	1	1
1.82	4.063	-0.333	-0.331	0.768	-0.763	-0.468	-0.468	0.848	-0.848	0.607	25	254	2	1	1
1.82	4.062	-0.48	-0.125	0.652	-0.852	-0.471	-0.461	0.827	-0.866	0.607	28	282	3	1	1
1.815	4.052	-0.392	-0.187	0.666	-0.881	-0.568	-0.251	0.766	-0.928	0.605	38	390	4	1	1
1.813	4.046	-0.341	-0.216	0.722	-0.787	-0.596	-0.187	0.753	-0.966	0.604	34	340	5	1	1
3.835	5.706	-0.599	0.198	0.366	0	-0.295	-0.41	0.705	0	1.197	22	220	1	1	2
3.578	5.325	-0.322	-0.343	0.775	-0.738	-0.468	-0.468	0.848	-0.848	1.193	26	260	2	1	2
3.577	5.322	-0.492	-0.09	0.609	-0.866	-0.471	-0.462	0.827	-0.866	1.192	26	260	3	1	2
3.568	5.31	-0.137	-0.276	0.541	-0.785	-0.568	-0.251	0.766	-0.927	1.189	25	250	4	1	2
3.563	5.302	-0.347	-0.222	0.742	-0.753	-0.596	-0.187	0.753	-0.966	1.188	22	220	5	1	2
6.194	6.913	-0.638	0.278	0.359	0	-0.294	-0.412	0.706	0	2.065	37	371	1	1	3
5.78	6.45	-0.484	-0.472	0.849	-0.836	-0.468	-0.468	0.848	-0.848	1.927	23	234	2	1	3
5.777	6.447	-0.558	-0.301	0.785	-0.866	-0.471	-0.462	0.827	-0.866	1.926	19	190	3	1	3
5.764	6.432	-0.6	-0.211	0.75	-0.926	-0.568	-0.251	0.766	-0.928	1.921	33	343	4	1	3
5.755	6.422	-0.547	-0.36	0.816	-0.905	-0.596	-0.187	0.753	-0.966	1.918	23	230	5	1	3
2.216	4.946	0.001	-0.997	0	0	0	-1	0.139	0	0.652	29	290	1	2	1
2.216	4.945	-0.153	-0.847	0	-0.174	-0.028	-0.972	0.123	-0.049	0.656	28	286	2	2	1
2.216	4.945	-0.153	-0.847	0	-0.173	-0.028	-0.972	0.123	-0.048	0.656	27	270	3	2	1

2.21	4.934	-0.088	-0.954	0	-0.157	-0.116	-0.94	0.072	-0.224	0.664	40	400	4	2	1
2.208	4.929	-0.069	-0.976	-0.012	-0.121	-0.095	-0.967	0.087	-0.187	0.661	34	340	5	2	1
4.356	6.48	-0.372	-0.101	0.316	0	0.019	-0.684	0.687	0	0.261	19	208	1	2	2
4.355	6.48	-0.224	-0.772	0.142	-0.334	-0.028	-0.972	0.123	-0.049	1.289	41	410	2	2	2
4.355	6.48	-0.199	-0.794	0.06	-0.32	-0.028	-0.972	0.123	-0.048	1.289	35	350	3	2	2
4.345	6.465	-0.195	-0.899	0.511	-0.356	-0.116	-0.94	0.072	-0.224	1.305	48	480	4	2	2
4.34	6.459	-0.08	-0.957	0.19	-0.245	-0.095	-0.967	0.087	-0.187	1.3	34	340	5	2	2
7.035	7.858	-0.141	-0.718	-0.38	0	-0.012	-0.977	0.132	0	2.083	24	287	1	2	3
7.034	7.85	-0.137	-0.863	0.04	-0.214	-0.026	-0.974	0.222	-0.045	2.083	31	325	2	2	3
7.034	7.85	-0.26	-0.736	0.301	-0.436	-0.028	-0.972	0.123	-0.049	2.082	39	390	3	2	3
7.018	7.832	-0.166	-0.914	0.325	-0.321	-0.116	-0.94	0.072	-0.224	2.076	37	370	4	2	3
7.011	7.824	-0.102	-0.959	0.194	-0.221	-0.095	-0.967	0.087	-0.187	2.099	37	370	5	2	3
2.232	4.982	0	-0.997	0	0	0	-1	0	0	0.744	25	250	1	3	1
2.232	4.982	0	-0.997	0	0	0	-1	0	0	0.744	25	250	2	3	1
2.232	4.982	0	-0.997	0	0	0	-1	0	0	0.744	25	250	3	3	1
2.232	4.982	0	-0.998	0	0	0	-1	0	0	0.744	25	250	4	3	1
2.232	4.982	0	-0.997	0	0	0	-1	0	0	0.744	24	240	5	3	1
4.387	6.528	0	-0.967	0	0	0	-1	0	0	1.462	20	200	1	3	2
4.387	6.528	0	-0.998	0	0	0	-1	0	0	1.462	29	290	2	3	2
4.387	6.528	0	-0.998	0	0	0	-1	0	0	1.462	29	290	3	3	2
4.387	6.528	0	-0.997	0	0	0	-1	0	0	1.462	26	260	4	3	2
4.387	6.528	0	-0.997	0	0	0	-1	0	0	1.462	26	260	5	3	2
7.086	7.908	0	-0.997	0	0	0	-1	0	0	2.362	26	260	1	3	3
7.086	7.908	0	-0.998	0	0	0	-1	0	0	2.362	27	270	2	3	3
7.086	7.908	0	-0.998	0	0	0	-1	0	0	2.362	27	270	3	3	3
7.086	7.908	0	-0.997	0	0	0	-1	0	0	2.362	27	270	4	3	3
7.086	7.908	0	-0.997	0	0	0	-1	0	0	2.362	27	270	5	3	3
2.202	4.922	-0.143	-0.715	0.713	0	-0.12	-0.76	0.531	0	0.411	24	263	1	4	1
2.143	4.783	-0.196	-0.8	-0.261	-0.344	-0.361	-0.639	0.346	-0.625	0.498	43	430	2	4	1
2.143	4.783	-0.231	-0.769	0.076	-0.401	-0.361	-0.639	0.346	-0.625	0.498	35	369	3	4	1
2.143	4.783	-0.432	-0.536	0.532	-0.723	-0.345	-0.664	0.357	-0.601	0.497	43	430	4	4	1
2.142	4.781	-0.433	-0.577	0.677	-0.756	-0.324	-0.702	0.371	-0.574	0.494	76	760	5	4	1
4.329	6.442	-0.06	-0.869	0.033	0	-0.176	-0.648	0.477	0	0.971	60	870	1	4	2
4.212	6.268	-0.243	-0.757	-0.032	-0.332	-0.361	-0.639	0.347	-0.625	0.972	26	279	2	4	2
4.212	6.268	-0.225	-0.775	-0.022	-0.348	-0.361	-0.639	0.346	-0.625	0.978	26	265	3	4	2
4.212	6.267	-0.146	-0.868	-0.065	-0.268	-0.345	-0.664	0.357	-0.601	0.976	28	280	4	4	2
4.21	6.265	-0.196	-0.822	-0.126	-0.359	-0.325	-0.701	0.371	-0.575	0.971	39	390	5	4	2
6.992	7.804	-0.191	-0.614	0.26	0	-0.176	-0.648	0.477	0	1.568	23	230	1	4	3
6.803	7.593	-0.207	-0.793	-0.089	-0.358	-0.36	-0.64	0.345	-0.624	1.546	21	210	2	4	3
6.803	7.593	-0.201	-0.798	-0.051	-0.346	-0.361	-0.639	0.347	-0.625	1.58	30	320	3	4	3
6.803	7.592	-0.16	-0.853	0.048	-0.317	-0.345	-0.664	0.357	-0.601	1.576	19	190	4	4	3
6.801	7.59	-0.208	-0.828	0.003	-0.379	-0.324	-0.702	0.371	-0.574	1.569	41	417	5	4	3

Comparing Table 7.4 and 7.10 it is clear that the optimal mass is less than or equal in the latter compared to the former. This result is to be expected as the above numerical examples exclude the strength (allowable laminate strain) constraints and hence the optimization problem is relaxed.

Again, structural mass savings are obtained when using an expanded set of ply orientations. Further, it is observed that the mass (M) of ply set 2 and ply set 5 vary by approximately 1%. This highlights that for aircraft structural design a discrete set of ply orientations can generally achieve most of the stiffness tailoring required and is comparable to a pseudo-continuous set. Additionally, it is observed that the optimal lamination parameters do not vary considerable over ply orientation sets 2-5. In the next sub-section, the DBM algorithm is under to determinate laminate stacking sequences for each example given in Table 7.10.

7.3.2 – Discrete Optimization using the DBM

For each of the sixty examples in Table 7.10, the optimum stacking sequence is determined using the DBM approach, assuming a laminate idealization as presented in Chapter 5. The results are presented in Table 7.11.

Table 7.11 – Optimal Lay-ups Obtained Using the Direct Branching Method

No.	*Stacking Sequence*	*Fitness*	*N*
1	$[45/45/45/90/45/45/90/90/45/90/45/45/90/90/45/90/45/45]_{MS}$	-0.014	18
2	$[60/60/60/60/60/60/60/60/60/60/60/60/0/0/0/0/0]_{MS}$	-0.046	17
3	$[60/60/60/60/60/60/60/60/60/60/60/75/-15/-15/-15/-15/-15]_{MS}$	-0.047	17
4	$[67.5/67.5/60/60/60/60/60/67.5/60/67.5/67.5/67.5/-15/-15/-15/-15/67.5]_{MS}$	-0.055	17
5	$[65/65/65/65/65/65/65/60/65/65/65/65/-10/-10/-10/-10/60]_{MS}$	-0.06	17
6	$[45/45/45/45/90/45/90/45/45/90/90/45/90/45/90/45/45/90/90/45/90/45/45]_{S}$	-0.023	23
7	$[60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/0/60/0/0/0/0/0]_{MS}$	-0.029	22
8	$[60/60/60/60/60/60/60/60/60/60/60/60/60/60/75/-15/-15/-15/60/-15/-15/-15]_{MS}$	-0.03	22
9	$[67.5/67.5/67.5/60/60/60/60/60/60/60/60/67.5/67.5/67.5/67.5/67.5/-15/-15/-15/-15/-15/-15]_{MS}$	-0.037	22
10	$[65/65/65/65/65/65/65/65/65/60/65/65/60/65/65/-10/60/-10/-10/-10/-10/-10]_{MS}$	-0.042	22
11	$[45/45/45/45/45/90/45/90/45/90/45/90/45/90/45/90/45/90/45/90/45/90/45/45/90/90/45/90]_{S}$	-0.038	28
12	$[60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/0/0/0/0/0/0/0]_{S}$	-0.023	26
13	$[60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/60/75/-15/-15/75/-15/-15/-15/-15/-15]_{S}$	-0.025	26
14	$[67.5/67.5/67.5/60/60/60/60/60/60/60/60/67.5/67.5/67.5/67.5/67.5/67.5/67.5/67.5/-15/-15/-15/-15/-15/-15/-15]_{S}$	-0.032	26
15	$[65/65/65/65/65/65/65/65/65/65/65/60/65/60/65/65/65/65/65/-10/-10/-10/-10/-10/-10/-10]_{S}$	-0.037	26
16	$[45/45/-45/-45/45/45/-45/-45/45/-45/45/-45/45/45/-45/-45/45/45/-45/45]_{S}$	-0.037	20
17	$[45/45/-45/-45/45/45/-45/-45/45/-45/45/-45/60/45/-45/-45/60/60/-45/-45]_{S}$	-0.038	20
18	$[45/45/-45/-45/45/45/-45/-45/45/-45/45/-45/60/45/-45/-45/60/60/-45/-45]_{S}$	-0.038	20
19	$[45/45/-45/-45/45/45/-45/-45/52.5/-45/52.5/-45/52.5/-45/52.5/52.5/-45/-45/52.5/-45]_{S}$	-0.041	20
20	$[45/45/-45/-45/50/-45/50/-45/50/50/-45/-45/50/-45/50/-45/50/-45/-45/50]_{S}$	-0.045	20
21	$[45/45/-45/45/-45/45/-45/45/-45/45/-45/-45/45/45/-45/-45/45/-45/45/45/-45/-45/45/-45/45/-45]_{S}$	-0.01	26
22	$[45/45/-45/45/-45/45/-45/45/-45/45/-45/-45/45/45/-45/-45/60/-45/60/-45/60/-45/60/60/-45/-45]_{S}$	-0.01	26
23	$[45/45/-45/45/-45/45/-45/45/-45/45/-45/-45/45/45/-45/-45/60/-45/60/-45/60/-45/60/60/-45/-45]_{S}$	-0.01	26
24	$[45/45/-45/45/-45/45/-45/52.5/-45/52.5/-45/-45/52.5/-45/52.5/-45/52.5/-45/52.5/-45/-45/52.5/52.5/-45/-45/52.5]_{S}$	-0.015	26
25	$[50/45/-45/45/-45/50/-45/50/-45/-45/50/-45/50/-45/50/-45/50/-45/50/-45/-45/50/50/-45/-45/-45]_{S}$	-0.019	26

26	[45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/45/-45/45/45]$_{MS}$	-0.01	32
27	[45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/60/-45/60/-45/60/-45/60/-45/-45/60/60/-45/-45]$_{MS}$	-0.011	32
28	[45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/45/-45/60/-45/60/-45/60/-45/60/-45/-45/60/60/-45/-45]$_{MS}$	-0.011	32
29	[45/45/-45/45/-45/45/-45/45/-45/52.5/-45/52.5/-45/52.5/-45/52.5/-45/52.5/-45/-45/52.5/-45/52.5/-45/52.5/-45/52.5/-45/52.5/52.5/-45/-45]$_{MS}$	-0.016	32
30	[50/45/-45/50/-45/50/-45/50/-45/50/-45/50/-45/-45/50/-45/50/-45/50/-45/50/-45/50/-45/50/-45/-45/50/-45/50/-45/-45]$_{MS}$	-0.021	32
31	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/-45]$_S$	-0.011	20
32	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/-45]$_S$	-0.011	20
33	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/45]$_S$	-0.011	20
34	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/45]$_S$	-0.011	20
35	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/45]$_S$	-0.011	20
36	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45]$_{MS}$	-0.045	27
37	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45]$_{MS}$	-0.045	27
38	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45]$_{MS}$	-0.045	27
39	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45]$_{MS}$	-0.045	27
40	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45]$_{MS}$	-0.045	27
41	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45]$_S$	-0.035	32
42	[45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45]$_S$	-0.035	32
43	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45]$_S$	-0.035	32
44	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45]$_S$	-0.035	32
45	[-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45/-45/45/45/-45/45/-45/-45/45]$_S$	-0.035	32
46	[45/45/45/45/45/-45/45/-45/45/-45/-45/45/-45/90/45/-45/90/-45/-45/90/-45/90/90/-45/90/90]$_{MS}$	-0.032	26
47	[60/60/60/60/60/60/-45/-45/-45/60/-45/-45/-45/-45/-45/60/-45/-45/-45/60]$_{MS}$	-0.085	20
48	[60/60/60/60/60/60/-45/-45/-45/60/-45/-45/-45/-45/-45/60/-45/-45/-45/60]$_{MS}$	-0.085	20
49	[60/60/60/60/52.5/52.5/-45/-45/-45/60/-45/-45/-45/60/-45/-45/-45/-45/-45/-45]$_{MS}$	-0.085	20
50	[60/60/60/55/55/55/-45/-45/-45/55/-45/-45/-45/-45/60/-45/-45/60/-45/-45]$_{MS}$	-0.087	20
51	[45/45/45/45/45/45/45/90/90/-45/45/-45/90/-45/-45/90/-45/-45/90/-45/-45/-45/90/-45/-45/-45]$_S$	-0.041	26
52	[60/60/60/60/60/60/60/-45/60/-45/-45/-45/60/-45/-45/-45/-45/60/-45/-45/60/-45/-45/-45/-45/-45]$_{MS}$	-0.075	26
53	[60/60/60/60/60/60/60/-45/60/-45/-45/-45/60/-45/-45/-45/-45/60/-45/-45/60/-45/-45/-45/-45/-45]$_{MS}$	-0.075	26
54	[60/60/60/60/60/60/52.5/-45/52.5/-45/-45/-45/60/-45/-45/-45/60/-45/-45/-45/60/-45/-45/-45/-45/-45]$_{MS}$	-0.076	26

55	*[60/60/60/60/55/55/55/-45/55/-45/-45/-45/55/-45/-45/55/-45/-45/-45/-45/-45/-45/60/-45/60/60]*$_{MS}$	-0.078	26
56	*[45/45/45/45/45/45/45/45/90/45/90/-45/90/-45/-45/90/-45/45/-45/-45/90/-45/-45/90/-45/-45/90/-45/-45/-45/45/45]*$_{MS}$	-0.041	32
57	*[60/60/60/60/60/60/60/60/60/-45/-45/60/-45/-45/-45/-45/60/-45/-45/-45/60/-45/-45/-45/-45/-45/-45/60/-45/-45/-45]*$_{MS}$	-0.018	31
58	*[60/60/60/60/60/60/60/60/60/-45/-45/60/-45/-45/-45/-45/60/-45/-45/-45/60/-45/-45/-45/-45/-45/-45/60/-45/-45/-45]*$_{MS}$	-0.018	31
59	*[60/60/60/60/60/60/60/52.5/52.5/-45/-45/60/-45/-45/-45/60/-45/-45/-45/-45/60/-45/-45/-45/-45/-45/-45/60/-45/60/-45]*$_{MS}$	-0.019	31
60	*[60/60/60/60/60/55/55/55/55/-45/-45/55/-45/-45/-45/55/-45/-45/-45/-45/60/-45/-45/-45/-45/60/-45/-45/-45/-45/-45]*$_{MS}$	-0.02	31

7.3.3 Discussion of Results

From Table 7.11, all of the constraints were satisfied indicated by a negative value for the constraint (this maps to a reserve factor >1). The number of function evaluations (stacking sequences which were evaluated by the fitness function) is calculated by multiplying the number of plies (where this equals the total number divided by two for a symmetric laminate) by the number of ply orientations in the utilized ply orientation set. Consequently, as the ply orientation set increases in size, so does the number of function evaluations. As such, the algorithm is generally less efficient for thick laminates or where the number of ply orientations is high. However, this is similar to a branch and bound approach which suffers from similar shortfalls. Nonetheless, if the objective is finding a lay-up which satisfies one constraint, the DBM will not only do this, but will also maximize the constraint reserve factor. Furthermore, the number of evaluations will be significantly less than then via enumeration or the branch and bound approach and can be determined a priori.

Concerning the buckling of long and thin anisotropic laminated composite plates, it is has been well documented in the literature that the outer layers of the laminate drive buckling resistence. Furthermore, with respect to $0, 90, \pm 45$ plies, ± 45 drive the buckling load. In contrast, using 60 degree plies leads to improved buckling performance. Consequently,using 60 degree plies in the design set increases the design space allowing for lower mass structures as well as improved buckling performance.

7.4 Conclusions

The objective of this chapter was to provide a number of examples to highlight the technical benefits introduced in this thesis. Initially, sixty examples were given showing the minimum mass for various combinations of loading conditions and sets of ply orientations. For each example, the constraints on the feasible region were calculated using the new method outlined in Chapter 2. By using lamination parameters and plate thickness as design variables, local optima were obtained using a gradient based method. In contrast, if ply orientations were used as design variables with a gradient approach, local optima may be obtained due to the resulting optimization problem being non-convex. It should be noted that in general, composite optimization is non-convex. For multi-part laminated composite optimization, the introduction of multiple thicknesses in lamination parameter results in a non-convex problem.

At the continuous level, it was shown that mass savings of circa 7% were achieved. This was where the aspect ratio was high (>3) and the loading was shear dominated and an expanded set of ply orientations was used. In particular, most weight savings were achieved using a set of $[0,90,\pm45,\pm30,\pm60]$ degree plies. Nonetheless, expanding this set to include all ply orientations varying by 5 degrees decreases (or retains) the value of the objective function. Furthermore, it was shown that using the restricted set of $[0,90,\pm45]$ was indeed appropriate for some loading conditions/geometry combinations. At the second level, the optimization was run using all four discussed approaches. It was shown that the ACO-DBM was the most efficient as highlighted in Chapter 5. In particular, it was observed that efficieny savings (measured in the number of iterations to determine a feasible design) of up to 40% could be obtained in comparison with techniques outlined in Chapter 4. Additionally, the performance of the SDGD was noted especially for smaller sets of ply orientations. The performance of the MPSO generally improved as the set of ply orientations increased. Noting the original PSO formulation was to serve continuous domains, the results from this Chapter support this notion. The GA performed the worse in both efficiency and quality of solution (relative to the other aforementioned approaches). In sum, the efficiency and functionality gains of the new methods/approaches detailed in this thesis have been demonstrated.

Finally, a number of numerical examples were provided to demonstrate the effectiveness of the DBM. The DBM successfully achieved quality designs in an a priori determined number of evaluations. The integration of the DBM with the ACO was shown in Chapter 5 and the in the numerical examples contained in this Chapter to be a powerful combination. Hence, the two-level optimization approach was proven to be successful in determining laminates of minimum thickness. In the next and final chapter. conclusions concerning this thesis are made and recommendations for future work are presented.

Chapter 8

Conclusions

In the previous chapter, a large number of numerical examples were presented. The results demonstrated the technical benefits outlined in this thesis and in particular the structural mass savings that could be achieved using an expanded set of ply orientations in lamination parameter space. In this chapter, a number of conclusions are made relating to the work presented. Initially, a summary of each chapter is presented with conclusions. After this, a discussion concerning contributions to the field of laminated composite optimization is given. Finally, suggestions for future work are outlined.

8.1 - Summary of Chapter Conclusions

In this section, the content and conclusions from each chapter are summarized.

Chapter 1

In Chapter 1, an introductory literature review was provided. Motivated by this review, the objectives for this thesis were outlined. That was, the derivation of an efficient, reliable, scalable and robust approach to the optimization of laminated composite structures using an expanded set of ply orientations and lamination parameters.

Chapter 2

In Chapter 2 a two-level method was presented to determine the feasible region of lamination parameters for any finite set of ply orientations, essentially solving half of a 20 year old problem (the remaining half being the derivation of the full feasible region with no restrictions on ply orientations). This breakthrough allows significant flexibility in laminated composite design. A mathematical proof was given to show the explicit relationships between the set of twelve lamination parameters. Note, any gradient based optimization using lamination parameters require the feasible region to maintain feasibility. To demonstrate the effectiveness of the approach, the complete feasible region of $0, 90, \pm 45, \pm 30, \pm 60$ degree plies was derived and the results presented in Chapter 3.

Chapter 3

In Chapter 3, the optimization solution was articulated. A two-level optimization approach is used. At the first level, lamination parameters and plate thickness are used as the design variables. The mass of the laminated composite plate is then minimized subject to buckling, strength (allowable laminate strain) and lamination parameter feasible region constraints. The first level determines the minimum thickness of the laminate for a given geometry and loading conditions. The optimal lamination parameters (along with the stiffness) determine the optimal stiffness. At the second level, ply orientations are used to determine a laminate stacking sequence which satisfy the buckling and strength constraints. As the thickness is determined at the first level, the number of plies is thus determined. Formally, the second level problem is a constraint satisfaction problem (CSP).

Chapter 4

In Chapter 4, the analyses of three population based meta-heuristic optimizers were given. In particular, genetic algorithms, particle swarm and ant colony optimization techniques were considered. Benchmarking was performed using a variety of loading conditions, plate aspect ratios and plate thicknesses. The analysis and numerical examples showed that an ant colony optimization (ACO) and particle swarm optimization (PSO) offer the best route to determining (efficiently) lay-ups which satisfies the set of constraints. Moreover, the ACO is more suited where the set of possible ply orientations is inherently discrete. In contrast, the PSO is better suited where the set of ply orientations is pseudo-continuous.

Chapter 5

Motivated by the findings of Chapter 4, three new optimization algorithms were presented in Chapter 5. In particular, a modified particle swarm optimization (MPSO), ant colony optimization - direct branching method (ACO-DBM) and stochastic discrete gradient descent (SDGD) were presented. The MPSO incorporates a new variant of the velocity function as well as a pseudo-gradient approach. Consequently, the swarm stochastically moves towards the global optima. The introduction of the direct branching method (DBM) was motivated by the fact that meta-heuristic approaches would often converge to good solutions (see Chapter 4), but needed slight refinement in order to solve the detailed constraint satisfaction problem. It was further proved that the DBM could be utilized to find the design which maximizes a single constraint reserve factor via the introduction of the concept of an idealized laminate. Combining the ACO and DBM therefore offers an attractive combination of a heuristic search with local refinement. Finally, the SDGD was introduced as a variant of the DBM incorporating discrete optimality criteria. The effectiveness of each new approach was demonstrated in Chapter 7.

Chapter 6

In Chapter 6, a conceptual approach for parallel optimization of multi-part laminated composite was presented. Building upon the work in the previous five chapters, a two-level approach was presented. At the first level, the structure, a simple four wall wingbox, was decomposed into plate elements. The mass of the structure was minimized subject to local design constraints using plate thicknesses and local lamination parameters. To increase the efficiency of the first level optimization, the sensitivity analysis was performed in parallel using a distributed computing approach. At the second level, the ACO-DBM was used to determine, in parallel, lay-ups which satisfy the set of local design constraints. The approach outlined in Chapter 6 potentially offers an efficient, reliable, scalable and robust approach to multi-part composite optimization.

Chapter 7

In Chapter 7, a number of numerical examples were presented. Using an expanded set of ply orientations, it was shown that mass savings of up to 7% were obtained for a simply supported symmetrically laminated composite plate under combined loading. It was further shown that for some loading conditions, the optimum set of ply orientations was in fact $0,90,\pm 45$ degrees, the common set adopted in industry. The analysis demonstrated that for different loading conditions, different sets of ply orientations may be appropriate. Furthermore, from an elastic tailoring perspective, almost any stiffness tailoring requirement in laminate design may be achieved from only a discrete set of ply orientations, namely between -85 and 90 degrees in 5 degree increments.

With respect to the discrete optimizers, the results showed that the MPSO had significantly better performance in comparison with the original PSO detailed in Chapter 4. It was also demonstrated that ACO-DBM was more efficient and obtained better than designs that the original ACO. Finally, the SDGD was successful in obtaining feasible solutions to the CSP but was observed to be generally less efficient that the MPSO and the ACO-DBM. Overall, it was shown that the ACO-DBM was the best route where an inherently discrete set of ply orientations was used leading to approximately 40% efficiency savings compared to the original variants outlined in Chapter 4.

8.2 - Contributions to the Field and Published Work

This thesis presents a number of developments to the field of laminated composite optimization. Initially, in Chapter 2, the formalization of an approach to determine the feasible region of lamination parameters for sets of discrete ply orientations was presented. Beyond this, a two-level optimization originally advocated in the literature was enhanced through the introduction of greater design scope and more efficient, reliable and robust second level optimization algorithms. In particular, an MPSO, ACO-DBM and SDGD were presented. All three represent new variations of their respective original formulations. The newer but rapidly growing field of parallel composite optimization was built upon in Chapter 6. In Chapter 6, for the first time, a conceptual two-level optimization approach using lamination parameters was presented. As such, all of the above represent developments in the field of laminated composite optimization. Additionally, the thesis has resulted in four journal publications (three published to date) and four conference papers. These details concerning these publications are outlined below.

Journal

Bloomfield, M.W., Diaconu, C.G., Weaver P.M., "On Feasible Regions of Lamination Parameters for Lay-Up Optimization of Laminated Composites", Proceedings of the Royal Society (A), 2009, v465(2104) pp. 1123-1143

Bloomfield, M.W., Herencia, J.E., Weaver, P.M., "Enhanced Two-Level Optimization of Anisotropic Laminated Composite Plates" Thin-Walled Structures, 2009, v47 pp.1161–1167

Bloomfield, M.W., Herencia, J.E., Weaver, P.M., "Benchmarking Optimization Techniques to Determine Laminate Stacking Sequences" Computers and Structures, 2010, v88(5-6), pp. 272-282

Bloomfield, M.W., Canale, G, Herencia, J.E., Weaver, P.M., "A Framework for Weight Minimization with Multiple Constraints in Laminated Composite Design" (submitted to Computers and Structures)

Conference

Bloomfield, M.W., Weaver, P.M., "Two-Level Optimization of Anisotropic Laminated Composite Plates" Aircraft Structural Design, Challenges for the Next Generation – Concept to Disposal, Royal Aeronautical Society, Liverpool UK, October 2008

Bloomfield, M.W., Weaver, P.M., "Two Level Optimization of Composite Plates Combining a Gradient Based and Ant Colony Approach" 7th ASMO UK ISSMO International Conference on Engineering Design Optimization, Bath UK, July 2008

Bloomfield, M.W., Herencia, J.E., Weaver, P.M.,, "Optimization of Anisotropic Composite Plates Using an Expanded Set of Ply Orientations" 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Chicago USA, April 2008

Weaver P.M., Bloomfield, M.W., "On the Potential for Elastic Tailoring for Layered Composites-Buckling Considerations" ICCM – Edinburgh, July 2009

The next sections concerns potential future work resulting from this thesis.

8.3 - Future Work

In the previous section, a summary of each chapter was given and conclusions were drawn. The work presented in this thesis has contributed to field of laminated composite design. The evolutionary nature of research assumes that undertaken work will allow future work to take place and build upon recent breakthroughs. In the area of laminated composite optimization, it is the author's opinion that the presented work can be developed in five key areas:

1. Determining the general feasible region of lamination parameters where ply orientations can take any continuous value
2. Derive ply continuity constraints as a function of lamination parameters for utilization at the first and second level of the optimization (allow multi-part laminated composite blending)
3. Develop closed form solutions for structural interaction to reduce the computational complexity and increase the efficiency of multi-part laminated composite design optimization
4. Enhance the ACO-DBM (or similar algorithm) to include ply continuity constraints
5. Perform full case studies for the parallel optimization, outlined in Chapter 6.

The first point (above) remains to the author's best knowledge, unsolved. Whilst approximations exist, due to the large numbers of constraints involved, they depend upon powerful (and commercially available) linear solvers such as CPLEX. Furthermore, more suitable approximations can be made, such as using the feasible region for a finite set of plies, [-90:5:90]. Knowledge of the full feasible region would allow for efficient optimization in laminated composite design as well as fully exploiting the design space for tow-steered fibre composites discussed in Chapter 1. In structural composites design, ply continuity is important to minimize local stresses between adjacent plies. However, this requirement induces significant complexity in the optimization and required additional work.

Note, Adams et al. (2003) attempted to solve the ply continuity problem (in multi-part composite structures) directly using a GA with migration. However, the formulation was in terms of ply orientations. Formulating the problem is terms of lamination parameters would potentially increase the efficiency of the optimization routine as well as exploiting the full design space and allow a full two-level approach. Additionally, enhancing the ACO-DBM (or other algorithm) to include additional constraints such as ply constraints at the second level of the parallel optimization would yield a more effective and industrially aligned approach. Finally, from a mathematical perspective, recent developments in numerical analysis in MINLP may allow for more efficient algorithms and possible the ability to determine global optima in continuous and multi-modal problems.

In summary, the objective of this thesis was the derivation (and presentation) of an efficient, reliable, scalable and robust approach to determine the stacking sequence of laminate composites of minimum weight which satisfy structural and design constraints. To solve this problem, a two-level approach was presented. At the first level, lamination parameters and plate thicknesses were used to minimize the mass of the laminate subject to a set of design constraints. Note, the first optimization is critically dependent upon the feasible region of lamination parameters expressed in terms of complex non-linear relationships between the parameters. At the second level, several discrete optimizers were used to efficiently determine a lay-up which satisfies the set of design constraints. The approach was then expanded to incorporate multi-part laminated composites. It is the authors opinion that the future of composite optimization is a mix of laminated composites as well as tow-steered (fibre placement) composites. Whilst this thesis has contributed to field of laminated composite design, it is hoped that it has provided some foundations that will allow progress in the field of tow-steered composite fibres. In summary, the work outlined in this thesis provides an efficient, reliable, robust approach for laminated composite optimization.

References

Adali, S., Walker, M., Verijenko, V.E. "Multiobjective optimization of laminated plates for maximum prebuckling, buckling and postbuckling strength using continuous and discrete ply angles", Composite Structures, 1996, v35(1) pp 117-130

Adams, D.B., Watson, L.T., Gurdel, Z., "Optimization and Blending of Composite Laminates Using Genetic Algorithms with Migration" MAMS, 2003, v10(3) pp.108-203

Antonio, C.A., "A hierarchical genetic algorithm with age structure for multimodal optimal design of hybrid composites" Struct Multidisc Optim, 2006, v31280–94.

Antonio, C.A., Torres, M.A., Soeiro A.V., "Optimization of laminated composite structures using a bi-level strategy", Composite Structures, 1995, v33 pp. 193–200.

Autio, M., "Determining the real lay-up of a laminate corresponding to optimal lamination parameters by genetic search", Struct Multidisc Optim, 2000, v20(4) pp. 301–10.

Aymerich, F., Serra, M., "An ant colony optimization algorithm for stacking sequence Design of Composite Laminates", CMES, 2006, v13(1) pp. 49-66

Aymerich, F., Serra, M., "Optimization of laminate stacking sequence for maximum buckling load using the ant colony optimization (ACO) metaheuristic", Composites Part A: Applied Science and Manufacturing, 2008, v39(2) pp 262-272

Barber, C. B., Dobkin, D. P. & Huhdanpaa, H. T. "The Quickhull algorithm for convex hulls" ACM Trans. Math. Softw. 1996 v22, 469–483. (doi:10.1145/235815.235821)

Bendsøe, M.P., Olhoff, N., Taylor, J.E., "A variational formulation for multicriteria structural optimization" J Struct Mech 1983, v11 pp. 523–44.

Bertsekas, D. P., Nedic, A. & Ozdaglar, A. E. "Convex analysis and optimization" 2003 Belmont, MA; Beijing, China: Athena Scientific and Tsinghua University Press

Boyd, S., *http://www.stanford.edu/~boyd/cvxbook/bv_cvxbook.pdf*

Bruyneel, M., Fleury, C., "Composite structures optimization using sequential convex programming" Adv Eng Software, 2002, v33 pp. 697–711

Canale, G., Weaver, P.M., "A simple model to evaluate aeroelastic tailoring of a wing box" Proceedings of RAeS Conference on Aircraft Structural Design, 2008, Liverpool, UK

Chen, J., Ge, R., Wei, J., "Probabilistic optimal design of laminates using improved particle swarm optimization" Eng Optim, 2008, v40(8) pp. 695–708.

CPLEX, 2010, http://www-01.ibm.com/software/integration/optimization/cplex/

Dantzig, G., "Notes on linear programming" 1953, Rand Corporation.

De Visser, J. A. P., "Aeroelastic and strength optimisation of a composite wing using a multilevel approach" AIAA-1999-1258, AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials St Louis, MO, Apr. 12-15, 1999, Collection of Technical Papers. Vol. 1 (A99-24601-05-39)

Diaconu, C. G., Sato, M. & Sekine, H. "Feasible region in general design space of lamination parameters for laminated composites" 2002a AIAA J. v40, 559–565. (doi:10.2514/2.1683)

Diaconu, C. G., Sato, M. & Sekine, H. "Buckling characteristics and layup optimization of long laminated composite cylindrical shells subjected to combined loads using lamination parameters" 2002b. Compos. Struct. v58, 423–433. (doi:10.1016 S0263-8223(02)00130-7)

Diaconu, C.G. and H. Sekine. "Layup Optimisation for Buckling of Laminated Composite Shells with Restricted Layer Angles". AIAA journal. 2004, v42(10) pp. 2153-2136

Dorigo., M., "Optimization, Learning and Natural Algorithms", PhD thesis, 1992, Politecnico di Milano, Italy

Engelbrecht, A.P., "Fundamentals of computational swarm intelligence", John Wiley and Sons Ltd, 2005

Farshi, B., Herasati, S., "Optimum weight design of fiber composite plates in flexure based on a two level strategy". Compos Struct, 2006, v73 pp. 495–504.

Fletcher, R., Powell, M. J. D., "A rapidly convergent descent method for minimization. Comput J, 1963, v6 pp.163–8.

Foldager, J., Hansen, J.S., Olhoff, N., "A general approach forcing convexity of ply angle optimization in composite laminates" Structural and Multidisciplinary Optimization 1998, v16(2-3), pp.201-211

Fukunaga, H., and Sekine, H., "Stiffness Design Method of Symmetric Laminates Using Lamination Parameters," AIAA Journal, 1992, v30(11) pp. 2791–2793.

Fukunaga, H., Sekine, H., Sato., M., Iino., "Buckling design of symmetrically laminated plates using lamination parameters" Computers & Structures, 1995, v57(4) pp. 643-649

Fukunaga, H., Vanderplaats, G., "Stiffness optimization of orthotropic laminated composites using lamination parameters". AIAA Journal, 1991, v29 pp. 641–646

Ghiasi, H., Pasini, D., Lassard, L., "Optimum stacking sequence design of composite materials Part I: Constant stiffness design" Composite Structures. 2009, v90(1) pp. 1-11

Grenestedt, J. L. & Gudmundson, P. "Lay-up optimisation of composite material structures". In Proc. IUTAM Symposium on Optimal Design with Advanced Materials 1993, pp. 311–336. Amsterdam, The Netherlands, Elsevier Science.

Grenestedt, J. L. "Layup optimisation against buckling of shear panels". Struct. Opt., 1991, v(3) pp.115-120

Gurdal, Z., Haftka, R.T., Hajela, P., "Design and optimization of laminated composite material" Wiley-Interscience, 1999

Gurdal, Z., Haftka, R.T., Nagendra, S., "Genetic algorithm for the design of laminated composite panels" SAMPE J, 1994, v30 pp. 207–18.

Gürdal, Z., Olmedo, R., "In-plane Response of Laminates with Spatially Varying Fiber Orientations: Variable Stiffness Concept," AIAA Journal, 1993, v31(4) pp.751-758.

Haftka, R. T., and Walsh, J. L., "Stacking Sequence Optimization for Buckling of Laminated Plates by Integer Programming," AIAA Journal, 1992, v30(2) pp. 814-819

Haftka, R.T, Gurdal, Z., "Elements of structural optimization" 3rd ed. Kluwer, 1992.

Hassan, B., Cohanim, A., Weck, O.,Venter, G., "Comparison of Particle Swarm Optimization and the Genetic Algorithm", AIAA-2005-1897 46th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference , Austin, Texas, 2005

Henderson, J.L., "Laminated plate design using genetic algorithms and parallel processing Computing Systems in Engineering" An International Journal, 1994, v5(4-6) pp. 441-453

Herencia, J. E., Weaver, P. M., Friswell, M. I., "Local optimisation of long anisotropic laminated fibre composite panels with T shape stiffeners". In Forty-seventh AIAA/ASME/AHS/ACS Structures, Structural Dynamics and Materials Conference, Newport, RI, 1–4 May, 2006

Herencia, J. E., Weaver, P. M., Friswell, M. I, "Local optimisation of anisotropic composite panels with T shape stiffeners" In Forty-eighth AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, Hawaii, 23–26 April, 2007a

Herencia, J.E., Weaver, P.M., and Friswell, M.I., "Local optimization of anisotropic composite panels with T shaped stiffeners including transverse shear effects and out-of-plane loading" Struct. Mul. Opt., 2007b, DOI:10.2007/s00158-008-0227-6

Herencia, J.E., Weaver, P.M., Friswell, M,I., "Optimization of Long Anisotropic Laminated Fiber Composite Panels with T-Shaped Stiffeners", AIAA Journal 2007c v45(10) pp. 2497-2509

Herencia, J.E,. Weaver, P.M., Friswell, M.I,, " Optimization of anisotropic composite panels with T shaped stiffeners including transverse shear effects and out-of-plane loading " Structural Multidisciplinary Optimization, 2008a, v37(2) pp. 165-184.

Herencia, J.E., Haftka, R. T., Weaver, P.M., Friswell, M.I., "Lay-Up Optimization of Composite Stiffened Panels Using Linear Approximations in Lamination Space", Technical Note, AIAA Journal, 2008b ,v46(9) pp. 2387-2391

Hirano, Y., Todoroki, A., "Stacking-sequence optimization of composite delta wing to improve flutter limit using fractal branch and bound method" JSME Int J A 2005, v48(2) pp.65–72.

Hirano, Y., Todoroki, A., "Stacking sequence optimizations for composite laminates using fractal branch and bound method (application for supersonic panel flutter problem with buckling load condition)". Adv Compos Mater, 2004, v13(2) pp. 89–107.

Hyer, M.W. Charette, R.F., "Use of curvilinear fiber format in composite structure design" AIAA Journal 1991, v 29(6) pp. 1011-1015

Hyer, M.W., Lee, H.H., "The use of curvilinear fiber format to improve buckling resistance of composite plates with central circular holes" Composite Structures, 1991, v18(3) pp. 239-61

IJsselmuiden, S., Abdalla, M., and Gürdal, Z., "Implementation of Strength Based Failure Criteria in the Lamination Parameter Design Space" AIAA-2007-2177 48th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Honolulu, Hawaii, Apr. 23-26, 2007

Kameyama, M., Fukunaga, H., "Optimum design of composite plate wing s for aeroelastic characteristics using lamination parameters" Comput Struct 2007, v 85(3 4) pp.213–24.

Kathiravan, R., Ganguli, R,. "Strength design of composite beam using gradient and particle swarm optimization" Compos Struct 2007, v81 pp.471–9.

Kennedy, J., Eberhart, R. C., "Particle swarm optimization" Proc. IEEE Int'l. Conf. on Neural Networks, IV, 1942–1948, 1995, Piscataway, NJ: IEEE Service Center.

Kere, P., Lento, J., "Design optimization of laminated composite structures using distributed grid resources", Compos Struct 2005, v71(3–4) pp.435–8.
Kim, T., Hwang I.H., "Optimal design of composite wing subjected to gust loads". Computers and Structures 2005, v83(19-20) pp. 1546-1554
Kogiso, N., Nakagawa, S., Murotsu, Y., "Lamination Parameters Applied to Reliability-Based In-Plane Strength Design of Composites", AIAA 2003 v40(11) pp. 2200-2207
Krog, L.A,. Olhoff, N., "Optimum topology and reinforcement design of disk and plate structures with multiple stiffness and eigenfrequency objectives", Computers and Structures, 1999, v72(4) pp. 535-563
Le Riche, R., Gaudin, J., "Design of dimensionally stable composites by evolutionary optimization" Compos Struct 1998 v41 pp.97–111.
Le Riche, R., Haftka, R.T., "Optimization of laminate stacking sequence for buckling load maximisation by genetic algorithm", AIAA 1993 v31(5), pp. 951-956
Le Riche, R., Haftka, R.T., "Improved genetic algorithm for minimum thickness composite laminate design", Compos Eng 1995, v5 pp. 143–61.
Lin, C.C., Lee, Y.J., "Stacking sequence optimization of laminated composite structures using genetic algorithm with local improvement" Compos Struct 2004, v63 pp. 339–45.
Liu, B., Haftka, R.T., Akgun, M., "Composite wing structural optimization using genetic algorithms and response surfaces" Struct Mult Opt, 2000 v20(2), pp.87-96
Liu, W., Butler, R., Mileham, A. R., and Green, A. J. "Bi-level optimization and postbuckling of highly strained composite stiffened panels". AIAA Journal 2006, v44(11), pp. 2562-2570
Liu, R. T. Haftka, M. A. Akgun, A. Todoroki, "Permutation genetic algorithm for stacking sequence design of composite laminates", Computer methods in applied mechanics and engineering, 2000, v186(2-4), pp. 357-372
Liu, B., Haftka, R.T., Trompette, P., "Maximisation of Buckling Loads of Composite Panels Using Flexural Lamination Parameters" Structural Multidsc. Optim. 2003. v26(1-2) pp. 28-36
Liu, D., Toropov, V., "Optimization of Blended Composite Wing Panels Using Smeared Stiffness Technique and Lamination Parameters" AIAA-2010-3079, In 51st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Orlando, Florida, 12–15 April, 2010
MATLAB software, The Mathworks, Inc., Natick, MA, USA, 1994-2009
Matsuzaki, R., Todoroki, A. "Stacking sequence optimization using fractal branch and bound method for unsymmetric laminates" Compos. Struct, 2007 v4 pp. 537–550.
Miki, M., Sugiyama, Y., "Optimum design of laminated composite plates using lamination parameters" AIAA J 1993, v31(5) pp.921–2.
Miki, M., "Material design of composite laminates with required in-plane elastic properties" In: Hayashi T, Kawata K, Umekawa S. editors. Progress in science and engineering of composites, ICCM-IV, Tokyo; 1982. pp. 1725–31.
Miki, M., "Material design of fibrous laminated composites with required flexural Stiffness" In: Mechanical behavior of materials-IV, proceeding of the fourth international conference; 1985. pp. 387–400.
Nagendra, S., Haftka, R.T., Gurdal, Z., "Design of blade stiffened composite panels by a genetic algorithm approach" In: Proceedings of the AIAA/ASME/AHS 34th

structures, structural dynamics and materials conference AIAA, Washington DC; 1993. pp. 2418–36.

Nagendra, S., Haftka, R.T., Gurdal, Z., "Stacking sequence optimization of simply supported laminates with stability and strain constraints" AIAA J 1992, v30(8) pp 2132–7.

Nagendra, S., Jestin, D., Gurdal, Z., Haftka, R.T., Watson, L.T., "Improved genetic algorithm for the design of stiffened composite panels", Computers & Structure, 1996 v58(3) pp. 543-555

Narita, Y., Hodgkinson, J.M., "Layerwise optimisation for maximising the fundamental frequencies of point-supported rectangular laminated composite plates" Compos Struct 2005, v69 pp.127–35.

Nelder, J.A., Mead, R., "Simplex method for function minimization" The Computer Journal, 1965, v7(4) pp.308–13.

Nemeth, M. P. "Importance of anisotropy on buckling of compression-loaded symmetric composite plates". AIAA Journal, 1986, v24(11) pp. 1831-1835.

Niu, C. Y. M. "Composite airframe structures-Practical design information and data" Hong Kong Conmilit Press Limited. 1992.

Nocedal, J., Wright, S.J., "Numerical optimization" 2nd ed. New York: Springer-Verlag; 2006.

Park, J.H., Hwang, J.H., Lee, C.S., Hwang, W., "Stacking sequence design of composite laminates for maximum strength using genetic algorithms" Compos Struct 2001 v52(2) pp.217–31.

Patel, H. D., "Multidisciplinary design optimization with superelements in MSC/NASTRAN".(MacNeal-Schwendler Corp., Los Angeles, CA) AIAA/USAF/NASA/OAI Symposium on Multidisciplinary Analysis and Optimization, 4th, Cleveland, OH, Sept. 21-23, 1992, Technical Papers. Pt. 1 (A93-20301 06-66), pp. 340-348.

Punch, W.F., Averill, R.C., Goodman, E.D., Lin, S.C., Ding, Y., Yip, Y.C., "Optimal design of laminated composite structures using coarse-grain parallel genetic algorithms" Comput Syst Eng, 1994 v5 pp.415–23.

Reddy, J. N., "Mechanics of Laminated Composite Plates and Shells: Theory and Analysis, Second Edition", CRC Press, 2003

Omkar, S.N. , Mudigerea, S.N., Narayana Naika D.G., Gopalakrishnana, S., "Vector evaluated particle swarm optimization (VEPSO) for multi-objective design optimization of composite structures", Computers & Structures v86 (1-2) 2008, pp. 1-14

Sendecky., J., "Longitudinal Shear Modulus of Filamentary Composite Containing Curvilinear fibers", Journal of Composite Materials, 1971 v5(1) pp. 82-93

Setoodeh, S., Abdalla, M.M., Gurdal, Z., "Design of variable – stiffness laminates using lamination parameters" Compos B: Eng, 2006, v37 pp. 301–9.

Setoodeh, S., Blom, A.W., Abdalla, M., Gurdal, Z, "Generating curvilinear fiber paths from lamination parameters distribution" Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, v 5, Collection of Technical Papers - 47th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials

Conference: 14th AIAA/ASME/AHS Adaptive Structures Conference, 8th AIAA Non-deterministic App, 2006, pp. 3440-3452

Setoodeh, S., Gurdal, Z., "Curvilinear fiber design of composite laminae by cellular automata" Collection of Technical Papers - AIAA/ASME/ASCE/AHS ASC Structures, Structural dynamics and Materials Conference, 2003, v7 pp. 5487-5495

Setoohdeh, S., Abdalla, M.M, Gürdal, Z., "Approximate Feasible Regions for Lamination Parameters" AIAA-2006-6973 11th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference, Portsmouth, Virginia, Sep. 6-8, 2006

Socha, K., Dorigo, M., "Ant colony optimization for continuous domains" Eur. J. Oper. Res., 2008, v185(3), pp.1155-1173

Soremekun, G., Gurdal, Z., Haftka, R.T., Watson, L.T., "Composite laminate design optimization by genetic algorithm with generalized elitist selection" Compos Struct 2001, v79 pp.131–43.

Suresh, S., Sujit, P.B., Rao, A.K., "Particle swarm optimization approach for multi-objective composite box-beam design", Composite Structures, 2007 v81 pp. 598-605

Svanberg, K., "The method of moving asymptotes – a new method for structural optimization", Int J Numer Meth Eng, 1987 v24 pp. 359–73.

Terada, Y., Todoroki, A., Shimamura, Y., "Stacking sequence optimizations using fractal branch and bound method for laminated composites" JSME Int J A, 2001, v44(4) pp. 490–498.

Thompson, B.B., Marks, R.J., El-Sharkawi, W.J., Miyamoto, R.T., "Inversion of underwater acoustic model for estimation of bottom parameters using modified particle swarm optimizer", Proceedings of the International Joint Conference on Neural Networks, 2003. pp. 1301-1306

Todoroki, A., Haftka, R.T., "Stacking sequence matching by a two stage genetic algorithm with consanguineous initial population. In: Proceedings of the 38th AIAA/ASME/ASCE/AHS/ASC structures, structural dynamics and materials conference part 2, 1997, pp. 1297–302.

Todoroki, A., Haftka, R.T., "Stacking sequence optimization by a genetic algorithm with a new recessive gene like repair strategy" Compos B: Eng 1998, v29 pp. 277–85.

Todoroki, A., Ishikawa, T. "Design of experiments for stacking sequence optimizations with genetic algorithm using response surface approximation" Compos Struct 2004, v64(3–4) pp. 349–57.

Todoroki, A., Sasada, N., Miki, M., "Object-oriented approach to optimize composite laminated plate stiffness with discrete ply angles" J Compos Mater 1996 v30(9) pp. 1020–41.

Todoroki, A., Sekishiro, M., "New iteration fractal branch and bound method for stacking sequence optimizations of multiple laminates" Compos Struct, 2007, v81 pp. 419–426.

Todoroki, A., Terada,Y., "Improved fractal branch and bound method for stacking-sequence optimizations of laminates" AIAA Journal, 2004, v42(1) pp. 141-148

Todoroki, A., Haftka, R.T., "Stacking sequence matching by two-stage genetic algorithm with consanguineous initial population" : Collection of Technical Papers - AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics & Materials Conference, v 2, 1997, p 1297-1302

Tsai, S. W. and Hahn, H. T., "Introduction to composite materials", 1980, Technomic Publishing Co., Inc., 750 Summer St., Stamford, CT. 06902
Tsai, S. W., Halpin, J. C. & Pagano, N. J., "Composite materials workshop", 1968 pp. 223–253 Stamford, CT: Technomic Publishing Co., Inc.
Vannucci, P., "On bending-tension coupling of laminates" 2001, Journal of Elasticity v 64(1) pp 13-28
Vannucci, P., "Plane anisotropy by the polar method" Meccanica, v 40, n 4-6 SPEC. ISS., December, 2005, pp 437-454
Vannucci, P., Verchery, G., "Stiffness design of laminates using the polar method, International Journal of Solids and Structures", 2001, v38(50-51) pp. 9281-94
Venkataraman S, Haftka R.T., "Optimization of composite panels – a review" In: Proceedings of the 14th annual technical conference of the American society of composites, Dayton, OH, September 27–29, 1999.
Venter, G., and Haftka, R.T., "Constrained particle swarm optimization using a bi-objective formulation", Structural and Multidiscipilanary Optimization, 2009 v40(1-6), pp. 65-76,
Vincenti, A.,Verchery, G., Vannucci, P., "Anisotropy and symmetry for elastic properties of laminates reinforced by balanced fabrics" Composites - Part A: Applied Science and Manufacturing, 2001, v32(10) pp. 1525-1532
Weaver P.M., "Designing composite structures: lay-up selection". Proceedings of the Institution of Mechanical Engineers – part G. J Aerospace Eng, v216; 2002. pp. 105–16.
Weaver, P. M., "Approximate analysis for buckling of compression loaded long rectangular plates with flexural/twist anisotropy", Philosophical Transactions of the Royal Society of London, Series A: Mathematical and Physical Sciences, 2006, v462(2065) pp. 59–74.
Weaver, P. M., "On optimization of long anisotropic flat plates subject to shear buckling loads", 45th AIAA/ASME/ASCE/AHS/ACS Structures, Structural Dynamics and Materials Conference, Palm Springs, CA, AIAA Paper 2004-2053, 2004.
Weaver, P.M., Nemeth, M.P., "Bounds on flexural properties and buckling response for symmetrically laminated plates", 2007, J. Engrg. Mech. 133, 1178
Weaver, P.M., Nemeth, M.P., "Improved design formulae for buckling of orthotropic plates under combined loading", 49th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Chicago, Illinois, 2008
Yamazaki, K. "Two-level optimisation technique of composite laminate panels by genetic algorithms". 1996, AIAA-96-1539-CP
Yang S, Wang M, Jiao L, "A Quantum Particle Swarm Optimization" In: Proceedings of CEC2004, the Congress on Evolutionary Computing, v(1) pp 320–324

Paginated blank pages are scanned as found in original thesis

No information is missing

Appendix A
Gradient Based Optimization

In non-linear constrained optimization, the general aim is to transform the problem into an easier sub-problem, which can then be solved iteratively. Recent solutions have focused on the utilization of the Karush-Kuhn-Tucker (KKT) equations (A.1-A.3). The KKT equations are necessary conditions for optimality in constrained optimization problems. If the problem is a so-called convex programming problem, that is, the objective and constraints are strict convex functions, then the KKT equations are both necessary and sufficient for global optima.

The gradient based approach articulated in this thesis utilizes a SQP approach based upon an active set (of constraints). The details of the SQP active set approach used in MATLAB is are now outlined. In non-linear programming, the objective function (*f*) is minimized subject to linear and non-linear constraints (*G*). A Lagrangian approach is adopted to handle the objective function, *f*, and the constraints (linear and non-linear, equality G^{eq} and inequality G^{in}). The Lagrangian, *L* is formulated as,

minimize
$$L = f + \sum_{i=1}^{m} \lambda_i G_i^{eq} + \sum_{j=1}^{n} \mu_j G_j^{in} \tag{A.1}$$

such that
$$\nabla L = \nabla f + \sum_{i=1}^{m} \lambda_i \nabla G_i^{eq} + \sum_{j=1}^{n} \mu_j \nabla G_j^{in} = 0 \tag{A.2}$$

and where (Primal feasibility condition),

$$\begin{aligned} G_i^{eq}(x) &= 0 \\ G_j^{in}(x) &<= 0 \end{aligned} \tag{A.3}$$

and (Dual feasibility condition),

$$\mu_j \geq 0 \tag{A.4}$$

Furthermore, the non-degeneracy condition is satisfied, if for all i

$$\nabla G_i \neq 0 \qquad where \qquad G_i = 0 \tag{A.5}$$

Additionally, the complementary slackness property states,

$$\mu_j G_j^{in}(x) = 0 \ \forall \ j \tag{A.6}$$

Eqns. (A.1-A.2) form the Karush-Kuhn-Tucker conditions (Bertsekas et al. 2003). As mentioned a gradient based approach is adopted.

The SQP handles the sub-problem of the objective and set of active constraints. Due to the non-linear nature of the Lagrangian, it is often approximated as

$$L(x_k + \Delta x) \approx L(x_k) + \nabla L(x_k)^T \Delta x + \frac{1}{2} \Delta x^T H \Delta x \tag{A.7}$$

The problem is then transformed into a quadratic programming (QP) problem (which is convex). For non-convex Lagrange functions, the chance of finding local optima still exists. However, for convex functions, the transformation shown in Eqn. (A.7) preserves the convexity. Note, in Eqn. (A.7) H is an approximation to the Hessian (due to computational efficiency issues in calculating), the matrix of second order derivatives. The matrix H is updated as follows,

$$H_{k+1} = H_k + \frac{q_k q_k^T}{q_k^T s_k} - \frac{H_k^T s_k^T s_k H_k}{s_k^T H_k s_k} \tag{A.8}$$

where

$$s_k = x_{k+1} - x_k \tag{A.9}$$

and,

$$q_k = \left(\nabla f(x_{k+1}) + \sum_{i=1}^{m} \lambda_i \nabla G_i^{eq}(x_{k+1}) + \sum_{j=1}^{n} \mu_j \nabla G_j^{in}(x_{k+1}) \right) - \left(\nabla f(x_k) + \sum_{i=1}^{m} \lambda_i \nabla G_i^{eq}(x_k) + \sum_{j=1}^{n} \mu_j \nabla G_j^{in}(x_k) \right)$$

(A.10)

Thus s_k can be seen as the change in the solution vector. Additionally, q_k is the difference between the gradient of the Lagrangian function between the kth and kth +1 iteration. The solution to the QP (defined by A.7) sub-problem produces a vector d_k, which is used to form a recursive step,

$$x_{k+1} = x_k + \alpha_k d_k \qquad \textbf{(A.11)}$$

The step length parameter α_k is determined in order to produce a sufficient decrease in the so called merit function used by *fmincon* in MATLAB (via a line search). The merit function is defined as,

$$\Psi(x) = f(x) + \sum_{i=1}^{m} r_i \cdot G_i^{eq} + \sum_{j=1}^{n} r_i \cdot \max(0, G_i^{in}(x))$$

(A.12)

where,

$$r_i = r(k+1)_i = \max_i \left\{ \lambda_i, \frac{(r_k)_i + \lambda_i}{2} \right\} \qquad \textbf{(A.13)}$$

Note, r_i is initially set to,

$$r_i = \frac{\|\nabla f(x)\|}{\|\nabla G_i(x)\|} \qquad \textbf{(A.11)}$$

Note, the procedure outlined above continues until the KKT conditions are satisfied. It is further observed that the merit function transforms the problem into a one-dimensional problem.

Appendix B

MATLAB Optimization Routines

In this appendix, a number of MATLAB codes are provided highlighting the core algorithms/methods/approaches demonstrated in this thesis.

First Level (Continuous Optimization)

```
global a
global b
global matid
global Nx
global Ny
global Nxy
global Mx
global My
global Mxy
global eat_x
global eat_y
global eat_xy
global eac_x
global eac_y
global eac_xy
global de
global n
global h
global tt


%----------------------------------------------------------------------
----
% INPUT & OUTPUT FILES
%----------------------------------------------------------------------
----

ipath='/home/mark';
iname='bsolhn';
opath=ipath;
oname=iname;

de = [-90:5:90];

ini = data';
a = ini(1);
b = ini(2);
h0 = ini(3);
xi1a0 = ini(4);
xi2a0 = ini(5);
xi3a0 = ini(6);
```

```
xi4a0 =ini(7);
xi1d0 = ini(8);
xi2d0 = ini(9);
xi3d0 = ini(10);
xi4d0 =ini(11);
matid = ini(12);
Nx = ini(13);
Ny = ini(14);
Nxy = ini(15);
Mx = ini(16);
My = ini(17);
Mxy = ini(18);
eat_x = ini(19);
eat_y = ini(20);
eat_xy = ini(21);
eac_x = ini(22);
eac_y = ini(23);
eac_xy = ini(24);
maxt = ini(25);
mint = ini(26);


nllpcons = gencons(pi/180*de);
[llpcons, bounds] = freg(1,pi/180*de);

bounds = round2(bounds,4);


nonlin = @(x)nlc(x,llpcons,nllpcons);

[x0]=[h0;xi1a0;xi2a0;xi3a0;xi4a0;xi1d0;xi2d0;xi3d0;xi4d0];

[LB]=[mint;min(bounds(:,1));min(bounds(:,2));min(bounds(:,3));min(bound
s(:,4));min(bounds(:,1));min(bounds(:,2));min(bounds(:,3));min(bounds(:
,4))];
[UB]=[maxt;max(bounds(:,1));max(bounds(:,2));max(bounds(:,3));max(bound
s(:,4));max(bounds(:,1));max(bounds(:,2));max(bounds(:,3));max(bounds(:
,4))];
options = optimset('LargeScale', 'off','MaxIter',80,'TolX',1e-10, ...
        'MaxFunEval',500000,'GradConstr','off','GradObj','off', ...
        'FunValCheck','on','Display','iter','DerivativeCheck','on', ...
        'TolCon',1e-10,'NonlEqnAlgorithm','gn');


[x,fval,exitflag,output,lambda,grad]=fmincon(@Obj3lampawh,x0,[],[], ...
                              [],[],[LB],[UB],nonlin,options);
```

First Level Constraints:

```
function [G, Geq] = nlc(x0, llpcons, nllpcons)

      global a
      global b
```

```
        global matid
        global Nx
        global Ny
        global Nxy
        global Mx
        global My
        global Mxy
        global eat_x
        global eat_y
        global eat_xy
        global eac_x
        global eac_y
        global eac_xy
        global maxt
        global mint


        desvar = x0';
        h = desvar(1);
        xi1a = desvar(2);
        xi2a = desvar(3);
        xi3a = desvar(4);
        xi4a = desvar(5);
        xi1d = desvar(6);
        xi2d = desvar(7);
        xi3d = desvar(8);
        xi4d = desvar(9);



        [G1] = lamcheck(xi1a,xi2a,xi3a,xi4a,0,0,0,0,xi1d,xi2d,xi3d,xi4d,
llpcons, nllpcons);

        [A,D] = ADlampn(matid,h,xi1a,xi2a,xi3a,xi4a, xi1d,xi2d,xi3d,
xi4d);

        [G2] = Gbuck_pw_closed_form(a,b,D,Nx,Ny,Nxy) ; % 4 edges simply
supported (Nx,Nxy)

        [G3] =
Gstrain1(h,A,D,Nx,Ny,Nxy,Mx,My,Mxy,eat_x,eat_y,eat_xy,eac_x,eac_y,eac_x
y);


        G=vertcat(G2,G3,G1);
        Geq=[];

end

function G =
lamcheck(xi1a,xi2a,xi3a,xi4a,xi1b,xi2b,xi3b,xi4b,xi1d,xi2d,xi3d,xi4d,
llpcons, nllpcons)


iplp=[xi1a xi2a xi3a xi4a 1 ];
oplp=[xi1d xi2d xi3d xi4d 1];
xA=[iplp];
```

```
xB=[0 0 0 0];
xD=[oplp];
G1=llpcons;
G11 = G1*iplp';
G12 = G1*oplp';
cons = nllpcons;
cons1=[cons(:,2:5) -cons(:,end)];
cons2=[cons(:,2:5) -cons(:,end-1)];
for i=1:length(cons)
ksq=cons(i,1);
cv(i,1)=(xA*cons1(i,1:5)')^4+3*ksq*(xB*cons1(i,1:4)')^2-
4*ksq*(xA*cons1(i,1:5)')*(xD*cons1(i,1:5)');
cv(i,2)=(xA*cons2(i,1:5)')^4+3*ksq*(xB*cons2(i,1:4)')^2-
4*ksq*(xA*cons2(i,1:5)')*(xD*cons2(i,1:5)');
end
G13 =[ cv(:,1) ; cv(:,2)];
G=[G11; G12; G13];
end
```

Second Level (Discrete) Optimization

Ant Colony Optimization

```
function [stack_seq, fval, k]=antcolony(n,de)

pr = 0.8;
maxit=200;
np =20;
tol = 0.008;
ants = zeros(np,n,maxit);
ants(:,:,1) = randsrc(np,n, de);
d(:,1) = antsfitness(ants,1);
[bestant,ind] = minnd(d);
bt(1)=bestant;


gant = ants(ind(1,1),:,ind(1,2));

if bestant<=tol
    stack_seq = gant;
    fval = bestant;
    k =1;
return
end

tabu =zeros(n, length(de));

for k=2:maxit

    tabu = ants_tabu(ants, d, tabu, k, bt);
    ants = nants(tabu, ants, k);

ants(1,:,k) = gant;

ch=randsrc(1,1,1:np);
```

```
ants(ch,:,k) = local_improve(ants(ch,:,k),n,de);




    d(:,k) = antsfitness(ants,k);
    [bestant,ind] = minnd(d);
    bt(k) = bestant;
        gant = ants(ind(1,1),:,ind(1,2));
    if min(bt)<=tol
    stack_seq = gant;
    fval =min(bt);
return
end

end

stack_seq = gant;
fval = bestant;
k = maxit;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%
function tabu = ants_tabu(ants, d, tabu, k, bt)
global de
global pr


%tabuold = tabu*(1-pr); %pheromone evaporation

x0=size(ants);

total_tabu = zeros(x0(1,2), length(de));

for i=1:x0(1,1);
    [lbest,id] = min(d(i,:));
    ants0 = ants(i,:,id);

    for j=1:x0(1,2);
        idx = find(ants0(1,j)==de);
        total_tabu(j,idx) = total_tabu(j,idx)+abs((1/lbest));
    end

end



   if ((k>10) && (round2(bt(k-1),3)==round2(bt(k-8),3)))
        total_tabu = total_tabu.*rand(x0(1,2), length(de));
   end

tabu = total_tabu;

total_fit = sum(tabu,2);
norm = repmat(total_fit,1,length(de));
```

```
tabu = tabu./norm;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%

function ants = nants(tabu, ants, k)
global de
x0 = size(ants);
a0 = x0(1,1)-5;

for i=1:x0(1,1)
    for j=1:x0(1,2)

        if i<a0
        prob = [de; tabu(j,:)];

        ants(i,j,k)=randsrc(1,1,prob);

        else
           ants(i,j,k)=randsrc(1,1,de);
        end



    end
end

[a, idx] = max(tabu,[],2);

ants(1,:,k) = de(idx);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function d = antsfitness(ants,k)
global tt
x0 = size(ants(:,:,k));

for i=1:x0(1,1)

    fi1 = ants(i,:,k);

    if mod(tt,2) ==0

        fi2 = fi1(end:-1:1);

    else
        fi20 = fi1(1,1:end-1);
        fi2 = fi20(end:-1:1);
    end

    fi = pi/180*[fi1 fi2];
    d(i)=bsmin3(fi1);
end
```

Constraint (CSP) Evaluation

```
function f=bsmin3(x0)
     global a
      global b
      global matid
      global Nx
      global Ny
      global Nxy
      global Mx
      global My
      global Mxy
      global eat_x
      global eat_y
      global eat_xy
      global eac_x
      global eac_y
      global eac_xy
      global h
     global tt


fi1=x0;


if mod(tt,2)==0

    fi2=fi1(end:-1:1);
    fi=pi/180*[fi1 fi2];
    z=[-1+2/length(fi):2/length(fi):1];
    lpams(1,:)=[xig(fi,z,0) xig(fi,z,2)];

else

    fi2=fi1(1,1:end-1);
    fi2=fi2(end:-1:1);
    fi=pi/180*[fi1 fi2];
    z=[-1+2/length(fi):2/length(fi):1];
    lpams(1,:)=[xig(fi,z,0) xig(fi,z,2)];
    end



      xi1a=lpams(1,1);
      xi2a=lpams(1,2);
      xi3a=lpams(1,3);
      xi4a=lpams(1,4);
      xi1d=lpams(1,5);
      xi2d=lpams(1,6);
      xi3d=lpams(1,7);
      xi4d=lpams(1,8);



    [A,D] = ADlampn(matid,h,xi1a,xi2a,xi3a,xi4a,xi1d,xi2d,xi3d, xi4d);
    [G1] =
Gstrain1(h,A,D,Nx,Ny,Nxy,Mx,My,Mxy,eat_x,eat_y,eat_xy,eac_x,eac_y,eac_x
y);
```

```
    [G2]= Gbuck_pw_closed_form(a,b,D,Nx,Ny,Nxy)';

    G=[G1' G2];
    f=max(G);
```

Direct Branching Method

```
function [stack_seq,fval] = local_improve(stackseq,n,de)

init = bsmin3(stackseq);

stack_seq=stackseq;
for i=1:n
    for j=1:length(de)
        stack_seq(1,i)=de(1,j);
        fit(j,:) = bsmin3(stack_seq);
    end

    [T,idx] = min(fit);
    stack_seq(1,i) = de(1,idx);
end

fval =bsmin3(stack_seq);
```

Classical and MPSO

```
function [stack_seq,fval, k]=pso(n,de,method)
global tt
global type

np=20; %swarm size
tol=0.008; % stopping criteria
maxit=200; % maximum number of iterations
pop=zeros(np,n,maxit);
vel=zeros(np,n,maxit);

vel(:,:,1)=randsrc(np,n,rand);
pop(:,:,1)=randsrc(np,n,de);


k=1;
for i=1:np
    fitness(i,k)=bsmin3(pop(i,:,k));
end


lbest=pop(:,:,1);
[gb, gb1]=minnd(fitness);

gbest=pop(gb1(1,1),:,gb1(1,2));

if gb1<tol
    stack_seq=gbest;
    fval =gb;
```

```
    return
end
gbestit(1,:)=[k, gb];

for k=2:maxit

   if method==1
   [pop, vel]=changepso(pop, vel, lbest, gbest, k, fitness, maxit);
   else
    pop=psoderv(pop, gbest, k, fitness);
   end

    for i=1:np
    fitness(i,k)=bsmin3(pop(i,:,k));

    [q,r]=minnd(fitness(i,:));

    lbest(i,:)=pop(r(1,1),:,r(1,2));
    end
    [gb1, gb2]=minnd(fitness);
    gbest=pop(gb2(1,1),:,gb2(1,2));


   if gb1<tol
       stack_seq=gbest;
       fval=gb1;
       return
   end
    gbestit(k,:)=[k gb1];

end

stack_seq=gbest;
fval=gb1;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%

function pop = psoderv(pop, gbest, k, fitness)
global de
beta=rand;

x0 = size(pop);

for i=1:x0(1,1)
    for j=1:x0(1,2)


vel(i,j,k)=randn*(gbest(1,j)-pop(i,j,k-1));

        xprime(1,j)=pop(i,j,k-1)+vel(i,j,k);

        xdprime(1,j)=pop(i,j,k-1)+beta*vel(i,j,k);


    end
```

```
    xprime=rem(xprime,180);
    xdprime=rem(xdprime,180);

t1=-(bsmin3(xdprime)-bsmin3(pop(i,:,k-1)))/beta;
t2=-(bsmin3(xprime)-bsmin3(pop(i,:,k-1)))/beta;
 t3=[t1 t2];

        [npos,I]=max(t3);

        if I==1
            pop(i,:,k)=corrector(xdprime,de);
            %pop(i,:,k)=xdprime;
        else
            pop(i,:,k)=corrector(xprime,de);
            %    pop(i,:,k)=xprime;
        end

end


if ((k>3) && (round2(min(fitness(:,k-1)),2)==round2(mean(fitness(:,k-
2)),2)))
pop(:,:,k)=pop(:,:,k-1).*rand_gen(-2,2,x0(1,1),x0(1,2));
end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%
function [pop, vel] = changepso(pop, vel, lbest, gbest, k, fitness,
maxit)
global de

x0 = size(pop);
wmax=0.9;
wmin=0.4;

w=wmax-((wmax-wmin)/maxit)*k;

c1=1.5;
c2=1.5;
r2=rand;
r1=rand;

for i=1:x0(1,1)
    for j=1:x0(1,2)

vel(i,j,k)=w*vel(i,j,k-1)+c1*r1*(gbest(1,j)-pop(i,j,k-
1))+c2*r2*(lbest(i,j)-pop(i,j,k-1));
%pop(i,j,k)=corrector(rem(vel(i,j,k)+pop(i,j,k-1),180),de);
pop(i,j,k)=rem(vel(i,j,k)+pop(i,j,k-1),180);
    end
end


if ((k>3) && (round2(min(fitness(:,k-1)),2)==round2(min(fitness(:,k-
2)),2)))

pop(:,:,k)=randsrc(x0(1,1),x0(1,2),de);
```

```
end
```

SDGD

```
function [stack_seq,fval,count] = steepest_grad_desc(n,de)

%steepest descent method for discrete variables of fixed laminate
thickness

fminold=1000;
fmin=100;
fval=2;
c1=0;
stack_seq=randsrc(1,n,de);
count=0;

%disp(sprintf('Iterations              CBest
StallGens'))


while c1<30

   if fval<0.008
       return
   end

   count=count+1;

    if fminold>fmin
        c1=0;
    else
        c1=c1+1;

    src=randsrc(1,1,1:n);
    stack_seq(1,src)=randsrc(1,1,de);
    end
    fminold =fmin;

    for i=1:n
        for j=1:length(de)
            stackseq=stack_seq;
            stackseq(1,i)=de(1,j);

            fit(i,j)=bsmin3(stackseq);

% stacks(i,:) =[ nearest_neighbour(de,stack_seq(1,i),-1) stack_seq(1,i)
nearest_neighbour(de,stack_seq(1,i),1)];
%
% for j=1:3
% stackseq = stack_seq;
% stackseq(1,i)=stacks(i,j);
% fit(i,j)=bsmin3(stackseq);
```

```
        end
    end


    [fmin,fminidx] = minnd(fit);

    fmin=round2(fmin,4);




    %disp(sprintf('%4d\t\t\t%.5g\t\t\t%5d', count, fmin,c1))


    stack_seq(1,fminidx(1,1)) = de(1,fminidx(1,2));


    fval=fmin;
%         plot(count,fmin,'b*')
%     hold on

end
```